Junos® OS

Class of Service Feature Guide (Routers and EX9200 Switches)

Modified: 2019-06-20

Copyright © 2019, Juniper Networks, Inc.

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Junos® OS Class of Service Feature Guide (Routers and EX9200 Switches)
Copyright © 2019 Juniper Networks, Inc. All rights reserved.

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 https://support.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

About the Documentation xxxvii
Documentation and Release Notes xxxvii
Using the Examples in This Manual xxxvii
Merging a Full Example xxxviii
Merging a Snippet xxxviii
Documentation Conventions xxxix
Documentation Feedback xli
Requesting Technical Support xli
Self-Help Online Tools and Resources xlii
Creating a Service Request with JTAC xlii

Part 1 Overview

Chapter 1 **Understanding How Class of Service Manages Congestion and Defines Traffic Forwarding Behavior 3**

Understanding How Class of Service Manages Congestion and Controls Service Levels in the Network 4
CoS Applications 5
CoS Standards 6
How CoS Applies to Packet Flow Across a Network 6
The Junos OS CoS Components Used to Manage Congestion and Control Service Levels 7
Mapping CoS Component Inputs to Outputs 11
Default Junos OS CoS Settings 14
Packet Flow Through the Junos OS CoS Process Overview 17
Packet Flow Within Routers Overview 19
Configuring Basic Packet Flow Through the Junos OS CoS Process 19
Define Classifiers 20
Apply Classifiers to Incoming Packets on Interfaces 21
Define Policers to Limit Traffic and Control Congestion 22
Define Drop Profiles 23
Assign Each Forwarding Class to a Queue 23
Define Schedulers 23
Define Scheduler Maps 24
Define CoS Header Rewrite Rules 24
Apply Scheduler Maps and Rewrite Rules to Egress Interfaces 25
Example: Classifying All Traffic from a Remote Device by Configuring Fixed Interface-Based Classification 25
Interface Types That Do Not Support Junos OS CoS 31

Copyright © 2019, Juniper Networks, Inc.

Part 2 Configuring Class of Service

Chapter 2 Assigning Service Levels with Behavior Aggregate Classifiers 35

Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic . . . 36
Default IP Precedence Classifier 39
Default DSCP and DSCP IPv6 Classifiers 41
Default MPLS EXP Classifier 42
Default IEEE 802.1p Classifier 43
Default IEEE 802.1ad Classifier 44
Default Aliases for CoS Value Bit Patterns Overview 45
Defining Aliases for CoS Value Bit Patterns 48
Configuring Behavior Aggregate Classifiers 51
Applying Behavior Aggregate Classifiers to Logical Interfaces 53
Example: Configuring and Applying a Default DSCP Behavior Aggregate Classifier 58
Example: Configuring Behavior Aggregate Classifiers 65
Understanding DSCP Classification for VPLS 75
Example: Configuring DSCP Classification for VPLS 76
Configuring Class of Service for MPLS LSPs 79
Class of Service for MPLS Overview 79
Configuring the MPLS CoS Values 80
Rewriting IEEE 802.1p Packet Headers with the MPLS CoS Value 82
Applying DSCP Classifiers to MPLS Traffic 83
Applying a DSCP Classifier to MPLS Packets on a Core-facing Interface 83
Applying a DSCP Classifier to MPLS Traffic for L3VPN/VPLS 85
Applying MPLS EXP Classifiers to Routing Instances 87
Configuring and Applying Custom MPLS EXP Classifiers to Routing Instances 88
Applying Global Classifiers and Wildcard Routing Instances 89
Applying Global MPLS EXP Classifiers to Routing Instances 90
Applying Classifiers by Using Wildcard Routing Instances 91
Verifying the Classifiers Associated with Routing Instances 93
Applying MPLS EXP Classifiers for Explicit-Null Labels 94

Chapter 3 Assigning Service Levels with Multifield Classifiers 97

Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields 97
Configuring Multifield Classifiers 98
Using Multifield Classifiers to Set Packet Loss Priority 101
Example: Configuring and Applying a Firewall Filter for a Multifield Classifier . . . 103
Example: Classifying Packets Based on Their Destination Address 109
Example: Configuring and Verifying a Complex Multifield Filter 111
Configuring a Complex Multifield Filter 111
Verifying a Complex Multifield Filter 113

Chapter 4 Controlling Network Access with Traffic Policing 115

Controlling Network Access Using Traffic Policing Overview 115
Congestion Management for IP Traffic Flows 116
Traffic Limits 116
Traffic Color Marking 117

Copyright © 2019, Juniper Networks, Inc.

Forwarding Classes and PLP Levels . . . 119
Policer Application to Traffic . . . 119
Effect of Two-Color Policers on Shaping Rate Changes . . . 120
Configuring Policers Based on Logical Interface Bandwidth . . . 122
Example: Limiting Inbound Traffic at Your Network Border by Configuring an Ingress Single-Rate Two-Color Policer . . . 124
Example: Performing CoS at an Egress Network Boundary by Configuring an Egress Single-Rate Two-Color Policer . . . 133
Example: Limiting Inbound Traffic Within Your Network by Configuring an Ingress Single-Rate Two-Color Policer and Configuring Multifield Classifiers . . . 142
Example: Limiting Outbound Traffic Within Your Network by Configuring an Egress Single-Rate Two-Color Policer and Configuring Multifield Classifiers . . . 155
Overview of Tricolor Marking Architecture . . . 171
Enabling Tricolor Marking and Limitations of Three-Color Policers . . . 173
Configuring and Applying Tricolor Marking Policers . . . 175
Defining a Tricolor Marking Policer . . . 175
Applying Tricolor Marking Policers to Firewall Filters . . . 178
Applying Firewall Filter Tricolor Marking Policers to Interfaces . . . 178
Example: Configuring and Applying a Single-Rate Tricolor Marking Policer . . . 179
Configuring Single-Rate Tricolor Marking . . . 180
Configuring Color-Blind Mode for Single-Rate Tricolor Marking . . . 181
Configuring Color-Aware Mode for Single-Rate Tricolor Marking . . . 181
Effect on Low PLP of Single-Rate Policer . . . 182
Effect on Medium-Low PLP of Single-Rate Policer . . . 182
Effect on Medium-High PLP of Single-Rate Policer . . . 183
Effect on High PLP of Single-Rate Policer . . . 183
Configuring Two-Rate Tricolor Marking . . . 184
Configuring Color-Blind Mode for Two-Rate Tricolor Marking . . . 184
Configuring Color-Aware Mode for Two-Rate Tricolor Marking . . . 185
Effect on Low PLP of Two-Rate Policer . . . 185
Effect on Medium-Low PLP of Two-Rate Policer . . . 186
Effect on Medium-High PLP of Two-Rate Policer . . . 186
Effect on High PLP of Two-Rate Policer . . . 186
Example: Configuring and Verifying Two-Rate Tricolor Marking . . . 187
Applying Firewall Filter Tricolor Marking Policers to Interfaces . . . 194
Example: Applying a Single-Rate Tricolor Marking Policer to an Interface . . 195
Policer Overhead to Account for Rate Shaping in the Traffic Manager . . . 195
Policer Overhead to Account for Rate Shaping Overview . . . 195
Example: Configuring Policer Overhead to Account for Rate Shaping . . . 196

Chapter 5 **Defining Forwarding Behavior with Forwarding Classes . . . 205**

Understanding How Forwarding Classes Assign Classes to Output Queues . . . 205
Output Queue Assignments Based on Forwarding Class . . . 206
Devices That Support Up to Four Forwarding Classes . . . 206
Devices That Support Up to 16 Forwarding Classes . . . 207
Default and Configurable Packet Loss Priority Values . . . 207

Copyright © 2019, Juniper Networks, Inc.

Configuration Statements Used to Configure and Apply Forwarding Classes . . . 207
Default Forwarding Classes . . . 208
Configuring a Custom Forwarding Class for Each Queue . . . 212
Configuring Up to 16 Custom Forwarding Classes . . . 214
Enabling Eight Queues on Interfaces . . . 217
Assigning Multiple Forwarding Classes and Default Forwarding Classes . . . 218
Examples: Configuring Up to 16 Forwarding Classes . . . 219
Classifying Packets by Egress Interface . . . 221
Forwarding Policy Options Overview . . . 223
Configuring CoS-Based Forwarding . . . 224
Example: Configuring CoS-Based Forwarding . . . 227
Example: Configuring CoS-Based Forwarding for Different Traffic Types . . . 230
Example: Configuring CoS-Based Forwarding for IPv6 . . . 230
Applying Forwarding Classes to Interfaces . . . 231
Understanding Queuing and Marking of Host Outbound Traffic . . . 232
Host Outbound Traffic Overview . . . 232
Routing Engine Sourced Traffic . . . 232
Distributed Protocol Handler Traffic . . . 232
Default Queuing and Marking of Host Outbound Traffic . . . 233
Configured Queuing and Marking of Host Outbound Traffic . . . 233
Configured Queuing and Marking of Outbound Routing Engine Traffic Only . . . 233
Forwarding Classes and Fabric Priority Queues . . . 234
Default Fabric Priority Queuing . . . 234
Overriding Default Fabric Priority Queuing . . . 234
Default Routing Engine Protocol Queue Assignments . . . 234
Assigning Forwarding Class and DSCP Value for Routing Engine-Generated Traffic . . . 237
Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets . . . 238
Changing the Default Queuing and Marking of Host Outbound Traffic . . . 241
Example: Configuring Different Queuing and Marking Defaults for Outbound Routing Engine and Distributed Protocol Handler Traffic . . . 242
Overriding the Input Classification . . . 250

Chapter 6 **Defining Output Queue Properties with Schedulers . . . 253**

How Schedulers Define Output Queue Properties . . . 254
Queue Scheduling Components . . . 255
Default Schedulers Overview . . . 257
Configuring Schedulers . . . 258
Configuring Scheduler Maps . . . 258
Applying Scheduler Maps Overview . . . 259
Applying Scheduler Maps to Physical Interfaces . . . 259
Configuring Traffic Control Profiles for Shared Scheduling and Shaping . . . 260
Configuring an Input Scheduler on an Interface . . . 263
Understanding Interface Sets . . . 264
Configuring Interface Sets . . . 264
Interface Set Caveats . . . 267

Copyright © 2019, Juniper Networks, Inc.

Configuring Internal Scheduler Nodes 268
Example: Configuring and Applying Scheduler Maps 269

Chapter 7 **Controlling Bandwidth with Scheduler Rates 273**

Oversubscribing Interface Bandwidth 273
Verifying Configuration of Bandwidth Oversubscription 279
Examples: Oversubscribing Interface Bandwidth 279
Configuring Scheduler Transmission Rate 282
Example: Configuring Scheduler Transmission Rate 284
Allocation of Leftover Bandwidth 284
Providing a Guaranteed Minimum Rate 285
Verifying Configuration of Guaranteed Minimum Rate 288
Example: Providing a Guaranteed Minimum Rate 289
PIR-Only and CIR Mode 290
Excess Rate and Excess Priority Configuration Examples 290
Controlling Remaining Traffic 295
Bandwidth Sharing on Nonqueuing Packet Forwarding Engines Overview 298
Configuring Rate Limits on Nonqueuing Packet Forwarding Engines 299
Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs 300
Example: Applying Scheduler Maps and Shaping Rate to DLCIs 308
Example: Applying Scheduling and Shaping to VLANs 312
Example: Limiting Egress Traffic on an Interface Using Port Shaping for CoS . . . 319
Configuring Input Shaping Rates for Both Physical and Logical Interfaces 326

Chapter 8 **Setting Transmission Order with Scheduler Priorities and Hierarchical Scheduling 327**

Priority Scheduling Overview 327
Strict-High Priority Configuration Overview 328
Platform Support for Priority Scheduling 329
Configuring Schedulers for Priority Scheduling 330
Associating Schedulers with Fabric Priorities 331
Example: Associating a Scheduler with a Fabric Priority 332
Hierarchical Class of Service Overview 333
Hierarchical Class of Service Network Scenarios 336
Services to Subscribers 336
Services to Businesses 336
Wireless Backhaul 336
Understanding Hierarchical Scheduling 337
Hierarchical Scheduling Terminology 337
Scheduler Node-Level Designations in Hierarchical Scheduling 338
Hierarchical Scheduling at Non-Leaf Nodes 339
Priority Propagation in Hierarchical Scheduling 340
Configuring Hierarchical Schedulers for CoS 342
Hierarchical Schedulers and Traffic Control Profiles 343
Example: Building a Four-Level Hierarchy of Schedulers 345
Configuring the Interface Sets 346
Configuring the Interfaces 346
Configuring the Traffic Control Profiles 347
Configuring the Schedulers 347
Configuring the Drop Profiles 348

Copyright © 2019, Juniper Networks, Inc.

Configuring the Scheduler Maps . . . 348
Applying the Traffic Control Profiles . . . 349

Chapter 9 **Controlling Congestion with Scheduler RED Drop Profiles and Buffers . . 351**

Managing Congestion Using RED Drop Profiles and Packet Loss Priorities . . . 351
Defining Packet Drop Behavior by Configuring RED Drop Profiles . . . 355
Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers . . . 359
Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows . . . 361
Mapping PLP to RED Drop Profiles . . . 363
Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size . . . 365
Configuring Large Delay Buffers for Slower Interfaces . . . 367
Configuring the Maximum Delay Buffer for NxDS0 Interfaces . . . 371
Example: Configuring Large Delay Buffers for Slower Interfaces . . . 374
Example: Configuring the Delay Buffer Value for a Scheduler . . . 375
Example: Configuring the Physical Interface Shaping Rate . . . 376
Complete Configuration . . . 377
Enabling and Disabling the Memory Allocation Dynamic per Queue . . . 378
Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy . . . 381
Example: Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy . . . 383

Chapter 10 **Altering Outgoing Packet Headers Using Rewrite Rules . . . 387**

Rewriting Packet Headers to Ensure Forwarding Behavior . . . 388
Applying Default Rewrite Rules . . . 389
Configuring Rewrite Rules . . . 391
Configuring Rewrite Rules Based on PLP . . . 392
Applying IEEE 802.1p Rewrite Rules to Dual VLAN Tags . . . 393
Example: Applying an IEEE 802.1p Rewrite Rule to Dual VLAN Tags . . . 394
Applying IEEE 802.1ad Rewrite Rules to Dual VLAN Tags . . . 395
Example: Applying an IEEE 802.1ad Rewrite Rule to Dual VLAN Tags . . . 395
Rewriting IEEE 802.1p Packet Headers with an MPLS EXP Value . . . 396
Setting IPv6 DSCP and MPLS EXP Values Independently . . . 398
Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel . . . 398
Setting Ingress DSCP Bits for Multicast Traffic over Layer 3 VPNs . . . 400
Applying Rewrite Rules to Output Logical Interfaces . . . 401
Rewriting MPLS and IPv4 Packet Headers . . . 403
Example: Rewriting MPLS and IPv4 Packet Headers . . . 405
Example: Simultaneous DSCP and EXP Rewrite . . . 407

Copyright © 2019, Juniper Networks, Inc.

Rewriting the EXP Bits of All Three Labels of an Outgoing Packet 407
Example: Rewriting the EXP Bits of All Three Labels of an Outgoing Packet 408
Defining a Custom Frame Relay Loss Priority Map 409
Example: Per-Node Rewriting of EXP Bits 410
Example: Rewriting CoS Information at the Network Border to Enforce CoS Strategies 411
Example: Remarking Diffserv Code Points to MPLS EXPs to Carry CoS Profiles Across a Service Provider's L3VPN MPLS Network 422
Example: Remarking Diffserv Code Points to 802.1P PCPs to Carry CoS Profiles Across a Service Provider's VPLS Network 446
Assigning Rewrite Rules on a Per-Customer Basis Using Policy Maps Overview 469
Configuring Policy Maps to Assign Rewrite Rules on a Per-Customer Basis 471

Chapter 11 **Altering Class of Service Values in Packets Exiting the Network Using IPv6 DiffServ 475**

Resources for CoS with DiffServ for IPv6 475
System Requirements for CoS with DiffServ for IPv6 476
Terms and Acronyms for CoS with DiffServ for IPv6 476
Default DSCP Mappings 477
Default Forwarding Classes 478
Juniper Networks Default Forwarding Classes 481
Roadmap for Configuring CoS with IPv6 DiffServ 483
Configuring a Firewall Filter for an MF Classifier on Customer Interfaces 484
Applying the Firewall Filter to Customer Interfaces 485
Assigning Forwarding Classes to Output Queues 485
Configuring Rewrite Rules 486
DSCP IPv6 Rewrites and Forwarding Class Maps 486
Applying Rewrite Rules to an Interface 487
Configuring RED Drop Profiles 488
Configuring BA Classifiers 488
Applying a BA Classifier to an Interface 489
Configuring a Scheduler 489
Configuring Scheduler Maps 490
Applying a Scheduler Map to an Interface 491
Example: Configuring DiffServ for IPv6 491
Configuration 491
Verification 501

Part 3 **Configuring Platform-Specific Functionality**

Chapter 12 **Configuring Class of Service on ACX Series Universal Metro Routers . . . 509**

CoS on ACX Series Routers Features Overview 510
Understanding CoS CLI Configuration Statements on ACX Series Routers 511
Configuring CoS on ACX Series Routers 513

Copyright © 2019, Juniper Networks, Inc.

Classifiers and Rewrite Rules at the Global, Physical and Logical Interface Levels Overview . . . 514
Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels . . . 515
Applying DSCP and DSCP IPv6 Classifiers on ACX Series Routers . . . 517
Schedulers Overview for ACX Series Routers . . . 517
Shared and Dedicated Buffer Memory Pools on ACX Series Routers . . . 518
CoS for PPP and MLPPP Interfaces on ACX Series Routers . . . 521
Limitations That are Common for CoS on PPP and MLPPP Interfaces . . . 521
Limitations for CoS on PPP Interfaces . . . 522
Guidelines for Configuring CoS on PPP and MLPPP Interfaces . . . 523
Limitations for CoS on MLPPP Interfaces . . . 524
CoS Functionalities for IPv4 Over PPP Interfaces . . . 525
CoS Functionalities for IPv4 Over MLPPP Interfaces . . . 527
CoS for NAT Services on ACX Series Routers . . . 531
Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic in ACX Series . . . 532
Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic in ACX Series . . . 533
Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface in ACX Series . . . 533
RED Drop Profiles Overview on ACX Series Routers . . . 535
Configuring RED Drop Profiles on ACX Series Routers . . . 536
Hierarchical Class of Service in ACX Series Routers . . . 536
Hierarchical Scheduling on the Physical Interface . . . 537
Traffic Control Profiles . . . 537
Schedulers . . . 538
Drop Profiles . . . 538
Scheduler Maps . . . 538
Applying the Traffic Control Profiles . . . 539
Subscriber Services . . . 539
Configuring hierarchical class of service for Layer 3 VPN Service . . . 539
Configuring hierarchical class of service for Layer 2 VPN (Ethernet Pseudowires) Service . . . 541
Configuring hierarchical class of service for VPLS Service . . . 541
Verifying the hierarchical class of service configurations . . . 542
Storm Control on ACX Series Routers Overview . . . 550

Chapter 13 **Configuring Class of Service on M Series Multiservice Edge Routers . . . 553**

CoS Features and Limitations on M Series and T Series Routers . . . 553
CoS Features and Limitations on M320 Routers with Enhanced III FPCs . . . 560
Packet Flow on Juniper Networks M Series Multiservice Edge Routers . . . 561
Incoming I/O Manager ASIC . . . 562
Internet Processor ASIC . . . 562
Outgoing I/O Manager ASIC . . . 563
Enhanced CFEB and CoS on M7i and M10i Routers . . . 563
Working Around Multifield Classifier Limitations on M Series Routers . . . 564
Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface . . . 565

Copyright © 2019, Juniper Networks, Inc.

Chapter 14 Configuring Class of Service on MX Series 3D Universal Edge Routers . . 567

Junos CoS on MX Series 5G Universal Routing Platforms Overview 567
CoS Features and Limitations on MX Series Routers 568
Packet Flow on MX Series 5G Universal Routing Platforms 570
Example of Packet Flow on MX Series 5G Universal Routing Platforms 573
Configuring and Applying IEEE 802.1ad Classifiers 575
Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels 576
Understanding Scheduling and Shaping of Traffic Routed to GRE Tunnels 576
Configuration Overview 577
Configuration Caveats 577
Example: Performing Output Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels 578
CoS-Based Interface Counters for IPv4 or IPv6 Aggregate on Layer 2 593
Enabling a Timestamp for Ingress and Egress Queue Packets 594
Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface 595
Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic 596
Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic . . 597
Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface 597

Chapter 15 Configuring Class of Service on PTX Series Packet Transport Routers . . 599

CoS Features and Limitations on PTX Series Routers 599
CoS Feature Differences Between PTX Series Packet Transport Routers and T Series Routers 601
Understanding Scheduling on PTX Series Routers 604
Output Queue Priorities Supported by the Junos OS CLI on PTX Series Routers 604
Scheduling Processes on PTX Series Routers 604
Strict-Priority and Scheduling Processes on PTX Series Routers 605
Understanding Virtual Output Queues on PTX Series Packet Transport Routers 607
Introduction to Virtual Output Queues on PTX Series Packet Transport Routers 607
VOQ Architecture 608
Round-Trip Time Buffering 608
VOQ Advantages 608
Does VOQ Change How I Configure CoS? 609
Understanding How VOQ Works on PTX Series Routers 611
Understanding the Components of the VOQ Process 611
Understanding the VOQ Process 612
Fabric Scheduling and Virtual Output Queues on PTX Series Routers 613

Copyright © 2019, Juniper Networks, Inc.

Understanding the Packet Forwarding Engine Fairness and Virtual Output Queue Process . . . 615
Handling Congestion . . . 616
Example: Configuring Excess Rate for PTX Series Packet Transport Routers . . . 617
Identifying the Source of RED Dropped Packets on PTX Series Routers . . . 624
Configuring Virtual LAN Queuing and Shaping on PTX Series Routers . . . 631
Example: Configuring Virtual LAN Queuing and Shaping in PTX Series Packet Transport Routers . . . 634
Example: Configuring Strict-Priority Scheduling on a PTX Series Router . . . 637
Understanding CoS CLI Configuration Statements on PTX Series Routers . . . 645

Chapter 16 **Configuring Class of Service on T Series Core Routers . . . 649**

CoS Features and Limitations on M Series and T Series Routers . . . 649
Packet Flow on Juniper Networks T Series Core Routers . . . 656
Incoming Switch Interface ASICs . . . 657
T Series Routers Internet Processor ASIC . . . 657
Queuing and Memory Interface ASICs . . . 658
Outgoing Switch Interface ASICs . . . 658
Identifying PICs Restricted to Four Queues on T Series Core Routers . . . 659
Managing Ingress Oversubscription at the PFE . . . 660
Configuring Traffic Class Maps to Manage Ingress Oversubscription . . . 661
Example: Configuring Traffic Class Maps . . . 665
Applying a Shaping Rate to Physical Interfaces Overview . . . 676
Configuring the Shaping Rate for Physical Interfaces . . . 676

Part 4 **Configuring Line Card-Specific and Interface-Specific Functionality**

Chapter 17 **Feature Support of Line Cards and Interfaces . . . 681**

CoS Features of the Router Hardware, PIC, MIC, and MPC Interface Families . . . 681
Scheduling on the Router Hardware, PIC, MIC, and MPC Interface Families . . . 681
Schedulers on the Router Hardware, PIC, MIC, and MPC Families . . . 682
Queuing Parameters for the Router Hardware, PIC, MIC, and MPC Interface Families . . . 683

Chapter 18 **Configuring Class of Service for Tunnels . . . 685**

CoS for Tunnels Overview . . . 685
Tunneling and BA Classifiers . . . 686
Configuring CoS for GRE and IP-IP Tunnels . . . 687
Example: Configuring CoS for GRE or IP-IP Tunnels . . . 688
Configuring a GRE Tunnel to Copy ToS Bits to the Outer IP Header . . . 695

Copyright © 2019, Juniper Networks, Inc.

Chapter 19 Configuring Class of Service on Services PICs 697

CoS on Services PICs Overview 697
Configuring CoS Rules on Services PICs 699
Configuring Match Conditions in a CoS Rule 700
Configuring Actions in a CoS Rule 702
Configuring Application Profiles 702
Configuring Reflexive and Reverse CoS Actions 703
Configuring CoS Rule Sets on Services PICs 705
Example: Configuring CoS Rules on Services PICs 705
Packet Rewriting on Services Interfaces 707
Multiservices PIC ToS Translation 707
Fragmentation by Forwarding Class Overview 708
Configuring Fragmentation by Forwarding Class 709
Configuring Drop Timeout Interval for Fragmentation by Forwarding Class 710
Example: Configuring Fragmentation by Forwarding Class 712
Allocating Excess Bandwidth Among Frame Relay DLCIs on Multiservices PICs 716
Configuring Rate Limiting and Sharing of Excess Bandwidth on Multiservices PICs 718

Chapter 20 Configuring Class of Service on IQ and Enhanced IQ (IQE) PICs 721

CoS on Enhanced IQ PICs Overview 721
Calculation of Expected Traffic on IQE PIC Queues 722
Excess Bandwidth Calculations Terminology 722
Excess Bandwidth Basics 722
Logical Interface Modes on IQE PICs 724
Default Rates for Queues on IQE PICs 728
Sample Calculations of Excess Bandwidth Sharing on IQE PICs 730
Configuring the Junos OS to Support Eight Queues on IQ Interfaces for T Series and M320 Routers 744
BA Classifiers and ToS Translation Tables 745
Configuring ToS Translation Tables 745
Configuring Hierarchical Layer 2 Policers on IQE PICs 751
Configuring Excess Bandwidth Sharing on IQE PICs 753
Configuring Rate-Limiting Policers for High Priority Low-Latency Queues on IQE PICs 759
Applying Scheduler Maps and Shaping Rate to Physical Interfaces on IQ PICs 761
Examples: Applying a Shaping Rate 762
Shaping Rate Calculations 762
Example: Applying a Shaping Rate to a Clear-Channel T1 Interface on a Channelized T1 IQ PIC 764

Copyright © 2019, Juniper Networks, Inc.

Example: Applying a Shaping Rate to a Clear-Channel E1 Interface on a Channelized E1 IQ PIC . . . 765
Examples: Applying a Scheduler Map and Shaping Rate to Physical Interfaces on IQ PICs . . . 766
Example: Applying a Scheduler Map and Shaping Rate to a DS0 Channel of a Channelized T1 Interface on a Channelized T1 IQ PIC . . . 767
Example: Applying a Scheduler Map and Shaping Rate to DS0 Channels of a Channelized E1 Interface on a Channelized E1 IQ PIC . . . 769
Applying a Scheduler Map and Shaping Rate to a Clear-Channel T3 Interface on a Channelized DS3 IQ PIC . . . 772
Applying a Scheduler Map and Shaping Rate to Fractional T1 Interfaces on a Channelized DS3 IQ PIC . . . 774
Applying a Scheduler Map and Shaping Rate to a DS0 Channel of a T1 Interface in a Channelized T3 Interface on a Channelized DS3 IQ PIC . . . 777
Applying Scheduler Maps to Chassis-Level Queues . . . 778
Applying Custom Schedulers to Packet Forwarding Component Queues . . 780
Examples: Scheduling Packet Forwarding Component Queues . . . 781
Example: Applying a Chassis Scheduler Map to a 2-Port IQ PIC . . . 781
Example: Configuring ATM CoS with a Normal Scheduler and a Chassis Scheduler . . . 783
Example: Configuring Two T3 Interfaces on a Channelized DS3 IQ PIC . . . 785
Example: Applying Normal Schedulers to Two T3 Interfaces . . . 786
Example: Applying a Chassis Scheduler to Two T3 Interfaces . . . 788
Assigning Default Frame Relay Rewrite Rule to IQE PICs . . . 790
Defining Custom Frame Relay Rewrite Rule on IQE PICs . . . 791

Chapter 21 **Configuring Class of Service on Ethernet IQ2 and Enhanced IQ2 PICs . . . 793**

CoS on Enhanced IQ2 PICs Overview . . . 794
CoS Features and Limitations on IQ2 and IQ2E PICs (M Series and T Series) . . 795
Differences Between Gigabit Ethernet IQ and Gigabit Ethernet IQ2 PICs . . . 796
Shaping Granularity Values for Enhanced Queuing Hardware . . . 798
Ethernet IQ2 PIC RTT Delay Buffer Values . . . 800
Configuring BA Classifiers for Bridged Ethernet . . . 801
Setting the Number of Egress Queues on IQ2 and Enhanced IQ2 PICs . . . 803
Configuring the Number of Schedulers per Port for Ethernet IQ2 PICs . . . 803
Applying Scheduler Maps to Chassis-Level Queues . . . 805
Applying Custom Schedulers to Packet Forwarding Component Queues . . 807
Examples: Scheduling Packet Forwarding Component Queues . . . 808
Example: Applying a Chassis Scheduler Map to a 2-Port IQ PIC . . . 808
Example: Configuring ATM CoS with a Normal Scheduler and a Chassis Scheduler . . . 810
Example: Configuring Two T3 Interfaces on a Channelized DS3 IQ PIC . . . 812
Example: Applying Normal Schedulers to Two T3 Interfaces . . . 813
Example: Applying a Chassis Scheduler to Two T3 Interfaces . . . 815
Configuring a Policer Overhead . . . 817
CoS for L2TP Tunnels on Ethernet Interface Overview . . . 818

Copyright © 2019, Juniper Networks, Inc.

Configuring CoS for L2TP Tunnels on Ethernet Interfaces . . . 819
Configuring LNS CoS for Link Redundancy . . . 820
Example: Configuring L2TP LNS CoS Support for Link Redundancy . . . 821
Configuring Shaping on 10-Gigabit Ethernet IQ2 PICs . . . 824
Configuring Per-Unit Scheduling for GRE Tunnels Using IQ2 and IQ2E PICs . . . 826
Understanding Burst Size Configuration on IQ2 and IQ2E Interfaces . . . 828
Configuring Burst Size for Shapers on IQ2 and IQ2E Interfaces . . . 829
Configuring a CIR and a PIR on Ethernet IQ2 Interfaces . . . 830
Example: Configuring Shared Resources on Ethernet IQ2 Interfaces . . . 832
Configuring and Applying IEEE 802.1ad Classifiers . . . 836
Configuring Rate Limits to Protect Lower Queues on IQ2 and Enhanced IQ2 PICs . . . 837
Simple Filters Overview . . . 839
Configuring a Simple Filter . . . 839

Chapter 22 **Configuring Class of Service on ATM Interfaces . . . 841**

CoS on ATM Interfaces Overview . . . 841
Enabling Eight Queues on ATM Interfaces . . . 842
Example: Enabling Eight Queues on ATM2 IQ Interfaces . . . 843
Copying the Packet Loss Priority to the CLP Bit on ATM Interfaces . . . 848
Configuring CoS for L2TP Tunnels on ATM Interfaces . . . 848
Configuring CoS for ATM2 IQ Virtual Circuit Tunnels . . . 850
Applying IEEE 802.1p BA Classifiers to Ethernet VPLS Over ATM . . . 851
Example: Combining Layer 2 and Layer 3 Classification on the Same ATM Physical Interface . . . 852
Applying Scheduler Maps to ATM Interfaces . . . 853
Configuring ATM Scheduler Support for Ethernet VPLS over ATM Bridged Interfaces . . . 855
Example: Configuring ATM Schedulers for Ethernet VPLS over ATM Bridged Interfaces . . . 857
Applying Scheduler Maps to Logical ATM Interfaces . . . 858
Configuring Linear RED Profiles on ATM Interfaces . . . 859
Configuring Virtual Circuit CoS Mode on ATM Interfaces . . . 859

Chapter 23 **Configuring Class of Service on SONET/SDH OC48/STM16 IQE PICs . . . 861**

CoS on SONET/SDH OC48/STM16 IQE PIC Overview . . . 862
Packet Classification on SONET/SDH OC48/STM16 IQE PICs . . . 864
Translation Table on SONET/SDH OC48/STM16 IQE PICs . . . 865
Configuring Translation Tables on SONET/SDH OC48/STM16 IQE PICs . . . 866
Example: Configuring CoS Value Translation Tables on SONET/SDH OC48/STM16 IQE PICs . . . 867
Scheduling and Shaping on SONET/SDH OC48/STM16 IQE PICs . . . 870
Configuring Scheduling, Shaping, and Priority Mapping on SONET/SDH OC48/STM16 IQE PICs . . . 872
Priority Mapping on SONET/SDH OC48/STM16 IQE PICs . . . 873
Example: Configuring Priority Scheduling on SONET/SDH OC48/STM16 IQE PICs . . . 874
Scaling for SONET/SDH OC48/STM16 IQE PICs . . . 876
Transmission Rate with Intelligent Oversubscription on SONET/SDH OC48/STM16 IQE PICs Overview . . . 877

Copyright © 2019, Juniper Networks, Inc.

Example: Configuring Transmit Rates That Add Up to More Than 100 Percent . . . 881
Example: Configuring Transmission Rate with Intelligent Oversubscription on SONET/SDH OC48/STM16 IQE PICs . . . 882
Example: Configuring a CIR and a PIR on SONET/SDH OC48/STM16 IQE Interfaces . . . 890
MDRR on SONET/SDH OC48/STM16 IQE PICs . . . 891
Configuring MDRR on SONET/SDH OC48/STM16 IQE PICs . . . 891
Example: Configuring MDRR on SONET/SDH OC48/STM16 IQE PICs . . . 891
WRED on SONET/SDH OC48/STM16 IQE PICs . . . 891
Configuring WRED on SONET/SDH OC48/STM16 IQE PICs . . . 891
Example: Configuring WRED on SONET/SDH OC48/STM16 IQE PICs . . . 892
Excess Bandwidth Sharing on SONET/SDH OC48/STM16 IQE PICs . . . 892
Configuring Excess Bandwidth Sharing on SONET/SDH OC48/STM16 IQE PICs . . . 892
Egress Rewrite on SONET/SDH OC48/STM16 IQE PICs . . . 892
Configuring Rewrite Rules on SONET/SDH OC48/STM16 IQE PIC . . . 892
Forwarding Class to Queue Mapping on SONET/SDH OC48/STM16 IQE PICs . . 893
Configuring Forwarding Classes on SONET/SDH OC48/STM16 IQE PIC . . . 893
Configuring Rate Limits on SONET/SDH OC48/STM16 IQE PICs . . . 894
Example: Configuring Rate Limits on SONET/SDH OC48/STM16 IQE PICs . . . 894

Chapter 24 **Configuring Class of Service on 10-Gigabit Ethernet LAN/WAN PICs with SFP+ . . . 897**

CoS on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview . . . 897
BA and Fixed Classification on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview . . . 898
Example: Configuring IEEE 802.1p BA Classifier on 10-Gigabit Ethernet LAN/WAN PICs . . . 899
DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC with SFP+ . . . 900
Configuring DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC . . . 902
Queuing on 10-Gigabit Ethernet LAN/WAN PICs Properties . . . 903
Mapping Forwarding Classes to CoS Queues on 10-Gigabit Ethernet LAN/WAN PICs . . . 904
Scheduling and Shaping on 10-Gigabit Ethernet LAN/WAN PICs Overview . . . 905
Example: Configuring Shaping Overhead on 10-Gigabit Ethernet LAN/WAN PICs . . . 906

Chapter 25 **Configuring Class of Service on Enhanced Queuing DPCs . . . 907**

Enhanced Queuing DPC CoS Properties . . . 908
Configuring Rate Limits on Enhanced Queuing DPCs . . . 910
Configuring WRED on Enhanced Queuing DPCs . . . 912
Configuring MDRR on Enhanced Queuing DPCs . . . 913
Configuring Excess Bandwidth Sharing . . . 916
Excess Bandwidth Sharing and Minimum Logical Interface Shaping . . . 916
Selecting Excess Bandwidth Sharing Proportional Rates . . . 916
Mapping Calculated Weights to Hardware Weights . . . 917
Allocating Weight with Only Shaping Rates or Unshaped Logical Interfaces . . . 918
Sharing Bandwidth Among Logical Interfaces . . . 919

Copyright © 2019, Juniper Networks, Inc.

Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs 920
Configuring Customer VLAN (Level 3) Shaping on Enhanced Queuing DPCs . . 922
Simple Filters Overview 924
Configuring Simple Filters on Enhanced Queuing DPCs 925
Configuring a Simple Filter 926

Chapter 26 **Configuring Class of Service on MICs, MPCs, and MLCs 929**

CoS Features and Limitations on MIC and MPC Interfaces 930
Dedicated Queue Scaling for CoS Configurations on MIC and MPC Interfaces Overview 931
Queue Scaling for MPCs 932
Managing Remaining Queues 933
Verifying the Number of Dedicated Queues Configured on MIC and MPC Interfaces 934
Scaling of Per-VLAN Queuing on Non-Queuing MPCs 935
Increasing Available Bandwidth on Rich-Queuing MPCs by Bypassing the Queuing Chip 940
Flexible Queuing Mode 941
Flexible Queuing Mode Overview 942
Upgrading non-HQoS MPCs to Support Flexible Queuing 943
Disabling Flexible Queuing for non-HQoS MPCs to Optimize Power Utilization 943
Multifield Classifier for Ingress Queuing on MX Series Routers with MPC 944
Example: Configuring a Filter for Use as an Ingress Queuing Filter 945
Ingress Queuing Filter with Policing Functionality 948
Understanding the Ingress Queuing Policing Filter 948
Example: Configuring a Filter for Use as an Ingress Queuing Policing Filter 949
Ingress Rate Limiting on MX Series Routers with MPCs 953
Rate Shaping on MIC and MPC Interfaces 954
Granularity of Rate Shaping on MIC and MPC Interfaces 954
Accounting for Layer 1 and Layer 2 Overhead in Egress Rate-Shaping Statistics 955
Per-Priority Shaping on MIC and MPC Interfaces Overview 955
Example: Configuring Per-Priority Shaping on MIC and MPC Interfaces 960
Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates 966
Example: Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates 967
Managing Traffic with Different Encapsulations 968
Managing Downstream Cell-Based Traffic 969
Traffic Burst Management on MIC and MPC Interfaces Overview 970
Guidelines for Configuring the Burst Size 970
How the System Calculates the Burst Size 971
Understanding Hierarchical Scheduling for MIC and MPC Interfaces 972
Scheduler Node Scaling for MIC and MPC Interfaces 973
Hierarchical Scheduling Priority Levels for MIC and MPC Interfaces 973
Guaranteed Bandwidth and Weight of an Interface Node on MIC and MPC Interfaces 973

Copyright © 2019, Juniper Networks, Inc.

Hierarchical Scheduling for MIC and MPC Interfaces in Oversubscribed PIR Mode . . . 974
Configuring Ingress Hierarchical CoS on MIC and MPC Interfaces . . . 975
Configuring a CoS Scheduling Policy on Logical Tunnel Interfaces . . . 977
Per-Unit Queuing and Hierarchical Queuing for MIC and MPC Interfaces . . . 978
Queuing Models Supported for MIC and MPC Interfaces . . . 978
Limited Scale Per-Unit Queuing MPCs . . . 979
Hierarchical Queuing MICs and MPCs . . . 979
Scheduler Node Levels for MIC and MPC Interfaces . . . 980
Scheduler Node Levels for Per-Unit Queuing MPCs . . . 980
Scheduler Node Levels for Hierarchical Queuing MICs and MPCs . . . 980
Managing Dedicated and Remaining Queues for Static CoS Configurations on MIC and MPC Interfaces . . . 981
Configuring the Maximum Number of Queues for MIC and MPC Interfaces . . . 981
Configuring Remaining Common Queues on MIC and MPC Interfaces . . . 982
Excess Bandwidth Distribution on MIC and MPC Interfaces Overview . . . 983
Bandwidth Management for Downstream Traffic in Edge Networks Overview . . . 984
Effective Shaping Rate . . . 984
Shaping Modes . . . 984
Byte Adjustments . . . 985
Relationship with Other CoS Features . . . 985
Scheduler Delay Buffering on MIC and MPC Interfaces . . . 986
Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs . . . 986
Drop Profiles on MIC and MPC Interfaces . . . 989
Drop Profiles on Enhanced Queuing MIC and MPC Interfaces . . . 989
Implicit Scaling of WRED Profiles . . . 989
Intelligent Oversubscription on MIC and MPC Interfaces Overview . . . 990
Jitter Reduction in Hierarchical CoS Queues . . . 991
Queue Jitter as a Function of the Maximum Number of Queues . . . 991
Default Maximum Queues for Hierarchical Queuing MICs and MPCs . . . 992
Shaping Rate Granularity as a Function of the Rate Wheel Update Period . . . 993
Example: Reducing Jitter in Hierarchical CoS Queues . . . 993
CoS on Ethernet Pseudowires in Universal Edge Networks Overview . . . 999
CoS Scheduling Policy on Logical Tunnel Interfaces Overview . . . 1000
Configuring CoS on an Ethernet Pseudowire for Multiservice Edge Networks . . . 1000
CoS for L2TP LNS Inline Services Overview . . . 1001
Guidelines for Applying CoS to the LNS . . . 1002
Hardware Requirements for Inline Services on the LNS . . . 1002
Configuring Static CoS for an L2TP LNS Inline Service . . . 1003
CoS on Circuit Emulation ATM MICs Overview . . . 1005
Configuring CoS on Circuit Emulation ATM MICs . . . 1006
Understanding IEEE 802.1p Inheritance push and swap from a Transparent Tag . . . 1008
Configuring IEEE 802.1p Inheritance push and swap from the Transparent Tag . . . 1009

Copyright © 2019, Juniper Networks, Inc.

CoS on Application Services Modular Line Card Overview 1011
CoS Implementation in HTTP Reverse Proxy Scenario 1012
CoS Implementation in Transparent Proxy Scenario 1013
CoS Implementation in Mixed-Mode Scenario 1013

Chapter 27 **Configuring Class of Service on Aggregated, Channelized, and Gigabit Ethernet Interfaces 1015**

Limitations on CoS for Aggregated Interfaces 1015
Policer Support for Aggregated Ethernet Interfaces Overview 1017
Understanding Schedulers on Aggregated Interfaces 1018
Examples: Configuring CoS on Aggregated Interfaces 1019
Hierarchical Schedulers on Aggregated Ethernet Interfaces Overview 1021
Configuring Hierarchical Schedulers on Aggregated Ethernet Interfaces 1022
Example: Configuring Scheduling Modes on Aggregated Interfaces 1023
Enabling VLAN Shaping and Scheduling on Aggregated Interfaces 1029
Example: Configuring Per-Unit Schedulers for Channelized Interfaces 1031
Applying Layer 2 Policers to Gigabit Ethernet Interfaces 1033
Examples: Applying Layer 2 Policers to a Gigabit Ethernet Interface 1034

Part 5 **Configuration Statements and Operational Commands**

Chapter 28 **Configuration Statements 1037**

action 1045
address (CoS on ATM Interfaces) 1046
adjust-minimum 1047
adjust-percent 1048
application-profile 1049
application-sets (Services CoS) 1050
applications (Services CoS) 1050
atm-options 1051
atm-policer 1053
atm-scheduler-map 1054
atm-service 1055
buffer-size (Schedulers) 1056
bypass-queuing-chip 1058
bytes (Dynamic Traffic Shaping) 1059
cbr 1060
cdvt 1061
cell-mode (Dynamic Traffic Shaping) 1062
class (CoS-Based Forwarding) 1063
class (Forwarding Classes) 1064
classification-override 1065
classifiers (Definition) 1066
classifiers (Logical Interface) 1067
classifiers (Physical Interface) 1068
classifiers (Routing Instance) 1069
class-of-service 1070
class-of-service (Protocols MPLS) 1071
code-point-aliases 1072
code-point 1073

Copyright © 2019, Juniper Networks, Inc.

code-points (CoS) . . . 1074
copy-plp-all . . . 1075
copy-tos-to-outer . . . 1075
copy-tos-to-outer-ip-header . . . 1076
copy-tos-to-outer-ip-header-transit . . . 1077
data (FTP) . . . 1077
default (CoS Host Outbound Traffic) . . . 1078
delay-buffer-rate . . . 1079
destination-address (Services CoS) . . . 1080
destination (Interfaces) . . . 1081
discard (Forwarding Class) . . . 1082
drop-probability (Interpolated Value) . . . 1083
drop-probability (Percentage) . . . 1084
drop-profile (Schedulers) . . . 1085
drop-profile-map (Schedulers) . . . 1086
drop-profiles . . . 1087
drop-timeout (Forwarding Class) . . . 1088
dscp (Services CoS) . . . 1089
dscp (CoS Classifiers) . . . 1090
dscp (CoS Interfaces) . . . 1091
dscp (Multifield Classifier) . . . 1092
dscp (Rewrite Rules) . . . 1093
dscp (Rewrite Rules on Physical Interface) . . . 1094
dscp-code-point (CoS Host Outbound Traffic) . . . 1095
dscp-ipv6 (CoS Rewrite Rules) . . . 1096
egress-policer-overhead . . . 1097
egress-shaping-overhead . . . 1098
enhanced (forwarding-class-accounting) . . . 1100
enhanced-priority-mode . . . 1103
epd-threshold . . . 1104
excess-bandwidth-share . . . 1105
excess-priority . . . 1106
excess-rate . . . 1107
excess-rate-medium-high . . . 1108
excess-rate-medium-low . . . 1109
excess-rate-high . . . 1110
excess-rate-low . . . 1111
exclude-queue-overhead-bytes . . . 1112
exp . . . 1113
exp-push-push-push . . . 1114
exp-swap-push-push . . . 1115
explicit-null-cos . . . 1116
fabric (Class-of-Service) . . . 1117
family (CoS on ATM Interfaces) . . . 1118
family (Multifield Classifier) . . . 1119
fill-level (Drop Profiles) . . . 1120
fill-level (Interpolated Value) . . . 1121
filter (Applying to an Interface) . . . 1122
filter (Applying to a Logical Interface) . . . 1123

Copyright © 2019, Juniper Networks, Inc.

filter (Configuring) . . . 1124
firewall . . . 1125
flexible-queuing-mode . . . 1126
flexible-vlan-tagging . . . 1127
forwarding-class (Services PIC Classifiers) . . . 1128
forwarding-class (ATM2 IQ Scheduler Maps) . . . 1129
forwarding-class (BA Classifiers) . . . 1130
forwarding-class (CoS Host Outbound Traffic) . . . 1131
forwarding-class (Forwarding Policy) . . . 1132
forwarding-class (Fragmentation) . . . 1133
forwarding-class (Interfaces) . . . 1134
forwarding-class (Multifield Classifiers) . . . 1134
forwarding-class (Restricted Queues) . . . 1135
forwarding-class-accounting . . . 1136
forwarding-class-default (Forwarding Policy) . . . 1137
forwarding-classes-interface-specific . . . 1138
(OBSOLETE) forwarding-classes (Class-of-Service) . . . 1139
forwarding-policy . . . 1140
fragment-threshold (Forwarding Class Maps) . . . 1141
fragmentation-map . . . 1142
fragmentation-maps . . . 1143
frame-mode (Dynamic Traffic Shaping) . . . 1144
frame-relay-de (Defining Loss Priority Maps) . . . 1145
frame-relay-de (Defining Loss Priority Rewrites) . . . 1146
from (Services CoS) . . . 1147
ftp (Services CoS) . . . 1148
guaranteed-rate . . . 1149
hierarchical-scheduler . . . 1150
high-plp-max-threshold . . . 1151
high-plp-threshold . . . 1152
host-outbound-traffic (Class-of-Service) . . . 1153
hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) . . . 1154
ieee-802.1 (Classifier on Physical Interface) . . . 1156
ieee-802.1 (Host Outbound Traffic) . . . 1157
ieee-802.1 (Rewrite Rules on Logical Interface) . . . 1158
ieee-802.1 (Rewrite Rules on Physical Interface) . . . 1159
ieee-802.1ad . . . 1160
import (Classifiers) . . . 1161
import (Rewrite Rules) . . . 1161
inet-precedence (Classifier on Physical Interface) . . . 1162
inet-precedence (CoS Classifiers) . . . 1163
inet-precedence (CoS Rewrite Rules) . . . 1164
inet-precedence (Rewrite Rules on Physical Interface) . . . 1165
inet6-precedence (CoS Rewrite Rules) . . . 1166
ingress-policer-overhead . . . 1167
ingress-queuing-filter . . . 1169
ingress-shaping-overhead . . . 1170
input-excess-bandwidth-share . . . 1171
input-policer . . . 1172

Copyright © 2019, Juniper Networks, Inc.

input-scheduler-map . . . 1173
input-shaping-rate (Logical Interface) . . . 1174
input-shaping-rate (Physical Interface) . . . 1175
input-three-color . . . 1176
input-traffic-control-profile . . . 1177
input-traffic-control-profile-remaining . . . 1178
interface-set (Ethernet Interfaces) . . . 1179
interface-set (Hierarchical Schedulers) . . . 1180
interfaces (CoS) . . . 1181
internal-node . . . 1183
interpolate . . . 1184
iq-policing-filter . . . 1185
irb . . . 1186
layer2-policer . . . 1187
linear-red-profile . . . 1188
linear-red-profiles . . . 1189
logical-bandwidth-policer . . . 1190
logical-interface-aggregate-statistics . . . 1191
logical-interface-policer . . . 1192
loss-priority (BA Classifiers) . . . 1194
loss-priority (Firewall Filter) . . . 1194
loss-priority (Rewrite Rules) . . . 1195
loss-priority (Scheduler Drop Profiles) . . . 1196
loss-priority (Simple Firewall Filter) . . . 1197
loss-priority-maps . . . 1198
loss-priority-maps (Assigning to an Interface) . . . 1199
loss-priority-rewrites . . . 1200
loss-priority-rewrites (Assigning to an Interface) . . . 1201
low-plp-max-threshold . . . 1202
low-plp-threshold . . . 1203
lsp-next-hop (CoS-Based Forwarding) . . . 1204
match-direction (Services CoS) . . . 1204
max-burst-size . . . 1205
max-queues . . . 1206
max-queues-per-interface . . . 1208
member-link-scheduler . . . 1210
mode (Layer 2 Tunneling Protocol Shaping) . . . 1211
multilink-class . . . 1212
next-hop (Class-Of-Service) . . . 1213
next-hop-map . . . 1214
no-fragmentation . . . 1215
non-lsp-next-hop . . . 1216
output-forwarding-class-map . . . 1216
output-policer . . . 1217
output-three-color . . . 1218
output-traffic-control-profile . . . 1219
output-traffic-control-profile-remaining . . . 1221
overhead-accounting . . . 1222
packet-timestamp . . . 1223

Copyright © 2019, Juniper Networks, Inc.

peak-rate . . . 1224
per-session-scheduler . . . 1225
per-unit-scheduler . . . 1226
plp-to-clp . . . 1227
policer (Configuring) . . . 1228
policing-action . . . 1230
policy-map . . . 1231
priority (ATM2 IQ Schedulers) . . . 1232
priority (Fabric Priority) . . . 1233
priority (Fabric Queues, Schedulers) . . . 1234
priority (Schedulers) . . . 1235
protocol (Host Outbound Traffic) . . . 1236
protocol (Rewrite Rules) . . . 1237
protocol (Schedulers) . . . 1238
queue (Global Queues) . . . 1239
queue (Restricted Queues) . . . 1240
queue-depth . . . 1241
queue-threshold . . . 1242
red-buffer-occupancy . . . 1244
reflexive | revert | reverse . . . 1245
restricted-queues . . . 1246
rewrite-rules (CoS Host Outbound Traffic) . . . 1247
rewrite-rules (Definition) . . . 1248
rewrite-rules (Interfaces) . . . 1249
rewrite-rules (Physical Interfaces) . . . 1251
routing-instances (CoS) . . . 1252
rtvbr . . . 1253
rule (Services CoS) . . . 1255
rule-set (Services CoS) . . . 1256
scheduler (Fabric Queues) . . . 1256
scheduler (Scheduler Map) . . . 1257
scheduler-map (Fabric Queues) . . . 1257
scheduler-map (Interfaces and Traffic-Control Profiles) . . . 1258
scheduler-map-chassis . . . 1259
scheduler-maps (For ATM2 IQ Interfaces) . . . 1260
scheduler-maps (For Most Interface Types) . . . 1261
schedulers (CoS) . . . 1262
schedulers (Interfaces) . . . 1263
services (CoS) . . . 1263
shaping . . . 1264
shaping-rate (Applying to an Interface) . . . 1265
shaping-rate (Schedulers) . . . 1267
shaping-rate (Oversubscribing an Interface) . . . 1269
shaping-rate-excess-high . . . 1271
shaping-rate-excess-low . . . 1273
shaping-rate-excess-medium-high . . . 1275
shaping-rate-excess-medium-low . . . 1276
shaping-rate-priority-high . . . 1277
shaping-rate-priority-low . . . 1279

Copyright © 2019, Juniper Networks, Inc.

shaping-rate-priority-medium . . . 1281
shaping-rate-priority-medium-low . . . 1282
shaping-rate-priority-strict-high . . . 1283
shared-bandwidth-policer (Configuring) . . . 1284
shared-instance . . . 1285
shared-scheduler . . . 1286
simple-filter (Applying to an Interface) . . . 1287
simple-filter . . . 1288
sip (Application Profile) . . . 1289
source-address (Services CoS) . . . 1290
strict-priority-scheduler . . . 1290
sustained-rate . . . 1291
syslog (Services CoS) . . . 1291
system-defaults . . . 1292
term (Firewall Filter) . . . 1293
term (Services CoS) . . . 1295
term (Simple Filter) . . . 1296
then (Services CoS) . . . 1297
three-color-policer (Applying) . . . 1298
three-color-policer (Configuring) . . . 1299
traffic-class (Tunnels) . . . 1301
traffic-class-map . . . 1302
traffic-class-map (Apply to Interface) . . . 1304
traffic-control-profiles . . . 1305
traffic-manager . . . 1306
translation-table . . . 1310
transmit-rate (Schedulers) . . . 1311
transmit-weight . . . 1313
transparent . . . 1313
tri-color . . . 1314
tunnel . . . 1315
tunnel-services (Chassis) . . . 1316
unit . . . 1318
vbr . . . 1319
vc-cos-mode . . . 1320
vci . . . 1321
video (Application Profile) . . . 1322
vlan-tag . . . 1323
vlan-tagging . . . 1324
voice (Application Profile) . . . 1326

Chapter 29 Operational Commands . . . 1327

show class-of-service classifier . . . 1328
show class-of-service code-point-aliases . . . 1331
show class-of-service drop-profile . . . 1333
show class-of-service fabric scheduler-map . . . 1337
show class-of-service fabric statistics . . . 1339
show class-of-service forwarding-table . . . 1342
show class-of-service forwarding-table classifier . . . 1346

Copyright © 2019, Juniper Networks, Inc.

show class-of-service forwarding-table classifier mapping 1348
show class-of-service forwarding-table drop-profile 1350
show class-of-service forwarding-table fabric scheduler-map 1352
show class-of-service forwarding-table rewrite-rule 1354
show class-of-service forwarding-table rewrite-rule mapping 1356
show class-of-service forwarding-table scheduler-map 1358
show class-of-service forwarding-table traffic-class-map 1360
show class-of-service fragmentation-map 1362
show class-of-service interface 1364
show class-of-service loss-priority-rewrite 1395
show class-of-service l2tp-session 1397
show class-of-service policy-map 1399
show class-of-service rewrite-rule 1401
show class-of-service routing-instance 1403
show class-of-service scheduler-hierarchy interface 1404
show class-of-service scheduler-map 1406
show class-of-service traffic-class-map 1409
show class-of-service translation-table 1411
show interfaces forwarding-class-counters 1416
show interfaces queue 1421
show interfaces voq 1468

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

List of Figures

Part 1 Overview

Chapter 1 **Understanding How Class of Service Manages Congestion and Defines Traffic Forwarding Behavior 3**

Figure 1: Packet Flow Across the Network 7
Figure 2: Packet Flow Through CoS-Configurable Components 8
Figure 3: Packet Flow Through CoS-Configurable Components 11
Figure 4: CoS Classifier, Queues, and Scheduler 18
Figure 5: Packet Flow Through CoS- Configurable Components 18
Figure 6: CoS Classifier, Queues, and Scheduler 19
Figure 7: Packet Flow Through CoS- Configurable Components 20
Figure 8: Fixed-Interface Classifier Scenario 27

Part 2 Configuring Class of Service

Chapter 2 **Assigning Service Levels with Behavior Aggregate Classifiers 35**

Figure 9: How a Classifier Works 36
Figure 10: Behavior Aggregate Classifier with Two Queues 61
Figure 11: Behavior Aggregate Classifier Scenario 61
Figure 12: Behavior Aggregate Classifier Scenario 66

Chapter 3 **Assigning Service Levels with Multifield Classifiers 97**

Figure 13: How a Classifier Works 98
Figure 14: Multifield Classifier Based on TCP Source Ports 104
Figure 15: Multifield Classifier Scenario 104

Chapter 4 **Controlling Network Access with Traffic Policing 115**

Figure 16: Network Traffic and Burst Rates 117
Figure 17: Single-Rate Two-Color Policer Scenario 127
Figure 18: Traffic Limiting in a Single-Rate Two-Color Policer Scenario 127
Figure 19: Single-Rate Two-Color Policer Scenario 136
Figure 20: Traffic Limiting in a Single-Rate Two-Color Policer Scenario 136
Figure 21: Single-Rate Two-Color Policer Scenario 145
Figure 22: Traffic Limiting in a Single-Rate Two-Color Policer Scenario 145
Figure 23: Multifield Classifier Based on TCP Source Ports 146
Figure 24: Single-Rate Two-Color Policer Scenario 158
Figure 25: Single-Rate Two-Color Policer Scenario 159
Figure 26: Traffic Limiting in a Single-Rate Two-Color Policer Scenario 159
Figure 27: Multifield Classifier Based on TCP Source Ports 160
Figure 28: Flow of Tricolor Marking Policer Operation 171
Figure 29: Tricolor Marking Sample Topology 187

Chapter 5 **Defining Forwarding Behavior with Forwarding Classes 205**

Copyright © 2019, Juniper Networks, Inc.

Figure 30: Customer-Facing and Core-Facing Forwarding Classes 215
Figure 31: Sample CoS-Based Forwarding 228

Chapter 7 Controlling Bandwidth with Scheduler Rates 273

Figure 32: Handling Remaining Traffic 296
Figure 33: Another Example of Handling Remaining Traffic 297
Figure 34: Port Shaping Scenario 320

Chapter 8 Setting Transmission Order with Scheduler Priorities and Hierarchical Scheduling 327

Figure 35: Hierarchical Scheduling Architecture 333
Figure 36: Hierarchical Schedulers and Priorities 342
Figure 37: Building a Scheduler Hierarchy 345

Chapter 9 Controlling Congestion with Scheduler RED Drop Profiles and Buffers . . 351

Figure 38: Discrete and Interpolated Drop Profiles 352
Figure 39: Discrete and Interpolated Drop Profiles 356

Chapter 10 Altering Outgoing Packet Headers Using Rewrite Rules 387

Figure 40: Packet Flow Across the Network 388
Figure 41: Rewriting CoS Information at the Network Border to Enforce CoS Strategies Scenario 413
Figure 42: MPLS Packet Structure 424
Figure 43: Rewriting CoS Information at the Network Border to Transit an MPLS Network Scenario 424
Figure 44: Typical VPLS Topology 447
Figure 45: VPLS with CoS Scenario 449

Chapter 11 Altering Class of Service Values in Packets Exiting the Network Using IPv6 DiffServ 475

Figure 46: Basic IPv6 DiffServ Topology 491
Figure 47: IPv6 DiffServ Configuration 492

Part 3 Configuring Platform-Specific Functionality

Chapter 13 Configuring Class of Service on M Series Multiservice Edge Routers 553

Figure 48: M Series Router Packet Forwarding Engine Components and Data Flow 562

Chapter 14 Configuring Class of Service on MX Series 3D Universal Edge Routers . . 567

Figure 49: MX Series Router Packet Forwarding and Data Flow 570
Figure 50: Packet Handling on the M Series and T Series Routers 572
Figure 51: Packet Handling on the MX Series Routers 572

Chapter 15 Configuring Class of Service on PTX Series Packet Transport Routers . . 599

Figure 52: Packet Flow Through CoS Components on PTX Series Routers 610
Figure 53: VOQ Components on PTX Series Routers 611
Figure 54: Virtual Output Queues on PTX Series Routers 612
Figure 55: Example of VOQ 613
Figure 56: Fabric Scheduling and Virtual Output Queues Process 614
Figure 57: Packet Forwarding Engine Fairness with Virtual Output Queue Process 615

Copyright © 2019, Juniper Networks, Inc.

Figure 58: Virtual Output Queuing on PTX Series Routers 625
Figure 59: Topology for Configuring Strict-Priority Scheduling on a PTX Series Router 638

Chapter 16 **Configuring Class of Service on T Series Core Routers 649**

Figure 60: T Series Router Packet Forwarding Engine Components and Data Flow 657
Figure 61: Configuring Traffic Class Maps on T4000 Router with Type 5 FPC . . 665

Part 4 Configuring Line Card-Specific and Interface-Specific Functionality

Chapter 18 **Configuring Class of Service for Tunnels 685**

Figure 62: CoS with a Tunnel Configuration 688

Chapter 21 **Configuring Class of Service on Ethernet IQ2 and Enhanced IQ2 PICs . . . 793**

Figure 63: Topology to Verify Link Redundancy Support for L2TP LNS CoS 822

Chapter 22 **Configuring Class of Service on ATM Interfaces 841**

Figure 64: Example Topology for Router with Eight Queues 843

Chapter 26 **Configuring Class of Service on MICs, MPCs, and MLCs 929**

Figure 65: Architecture for MIC and MPC Interface Per-Priority Shaping 958
Figure 66: Scheduling Hierarchy for Per-Priority Shaping 959
Figure 67: Example of MIC and MPC Interface Scheduling Hierarchy 961
Figure 68: Sample Network Topology for Downstream Traffic 967
Figure 69: Sample Burst Shaping Rates 970
Figure 70: Scheduler Node Levels for Per-Unit Queuing MPCs 980
Figure 71: Scheduler Node Levels for Interfaces on Hierarchical Queuing and Scheduling MICs and MPCs 981
Figure 72: Processing of CoS Parameters in an L2TP LNS Inline Service 1002

Chapter 27 **Configuring Class of Service on Aggregated, Channelized, and Gigabit Ethernet Interfaces 1015**

Figure 73: Scaled Mode for Aggregated Ethernet Interfaces 1027
Figure 74: Replicated Mode for Aggregated Ethernet Interfaces 1028
Figure 75: Aggregated Ethernet Primary and Backup Links 1030

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

List of Tables

About the Documentation xxxvii

Table 1: Notice Icons xxxix
Table 2: Text and Syntax Conventions xl

Part 1 Overview

Chapter 1 Understanding How Class of Service Manages Congestion and Defines Traffic Forwarding Behavior 3

Table 3: CoS Mappings—Inputs and Outputs 11

Part 2 Configuring Class of Service

Chapter 2 Assigning Service Levels with Behavior Aggregate Classifiers 35

Table 4: Default IP Precedence (ipprec-compatibility) Classifier 40
Table 5: Default IP Precedence (ipprec-default) Classifier 40
Table 6: Default DSCP and DSCP IPv6 Classifiers 41
Table 7: Default MPLS EXP Classification 42
Table 8: Default IEEE 802.1p Classifier 43
Table 9: Default IEEE 802.1ad Classifier 44
Table 10: Default CoS Value Aliases 45
Table 11: Logical Interface Classifier Combinations 54
Table 12: Sample ba-classifier Loss Priority Assignments 66
Table 13: Default VPLS Classifiers 75
Table 14: MPLS CoS Values 82
Table 15: Default MPLS EXP Classifier 87

Chapter 4 Controlling Network Access with Traffic Policing 115

Table 16: Policer Actions 118
Table 17: Devices Versus TCM 174
Table 18: Tricolor Marking Policer Statements 176
Table 19: Color-Blind Mode TCM Color-to-PLP Mapping 181
Table 20: Color-Aware Mode TCM PLP Mapping 182
Table 21: Color-Blind Mode TCM Color-to-PLP Mapping 184
Table 22: Color-Aware Mode TCM Mapping 185

Chapter 5 Defining Forwarding Behavior with Forwarding Classes 205

Table 23: Default Forwarding Classes 208
Table 24: Sample Forwarding Class-to-Queue Mapping 215
Table 25: Default Queue Assignments for Packets Generated by the Routing Engine 235

Chapter 6 Defining Output Queue Properties with Schedulers 253

Copyright © 2019, Juniper Networks, Inc.

Table 26: Output Queue Scheduler Components 255
Table 27: Sample diffserv-cos-map Scheduler Mapping 269

Chapter 7 Controlling Bandwidth with Scheduler Rates 273

Table 28: Bandwidth and Delay Buffer Allocations by Configuration Scenario . . 277
Table 29: Bandwidth and Delay Buffer Allocations by Configuration Scenario . . 288
Table 30: Current Behavior with Multiple Priority Levels 290
Table 31: Current Behavior with Same Priority Levels 291
Table 32: Current Behavior with Strict-High Priority 291
Table 33: Strict-High Priority with Higher Load 292
Table 34: Sharing with Multiple Priority Levels 292
Table 35: Sharing with the Same Priority Levels 292
Table 36: Sharing with at Least One Strict-High Priority 293
Table 37: Sharing with at Least One Strict-High Priority and Higher Load 293
Table 38: Sharing with at Least One Strict-High Priority and Rate Limit 294
Table 39: Fine-Grained Queuing and Scheduling Support by Interface or PIC Type 303
Table 40: Fine-Grained Queuing and Scheduling Support by MIC or MPC Type 305

Chapter 8 Setting Transmission Order with Scheduler Priorities and Hierarchical Scheduling 327

Table 41: Scheduling Priority Mappings by FPC Type 330
Table 42: Resources for Learning More About HCoS 335
Table 43: Node Levels Designations in Hierarchical Scheduling 338
Table 44: Queue Priority 340
Table 45: Internal Node Queue Priority for CIR Mode 341
Table 46: Internal Node Queue Priority for PIR-Only Mode 341

Chapter 9 Controlling Congestion with Scheduler RED Drop Profiles and Buffers . . 351

Table 47: Buffer Size Temporal Value Ranges by Routing Device Type 366
Table 48: Recommended Delay Buffer Sizes 367
Table 49: Maximum Delay Buffer with q-pic-large-buffer Enabled by Interface 368
Table 50: Delay-Buffer Calculations 369
Table 51: NxDS0 Transmission Rates and Delay Buffers 372

Chapter 10 Altering Outgoing Packet Headers Using Rewrite Rules 387

Table 52: Default Packet Header Rewrite Mappings 390
Table 53: Default MPLS EXP Rewrite Rules 404

Chapter 11 Altering Class of Service Values in Packets Exiting the Network Using IPv6 DiffServ 475

Table 54: Default DSCP Mappings 477
Table 55: Default Behavior Aggregate Classification 479
Table 56: Forwarding Classes and Queues Classification 480
Table 57: Resources Assigned to Queues 481
Table 58: Default Forwarding Classes 482

Part 3 Configuring Platform-Specific Functionality

Chapter 13 Configuring Class of Service on M Series Multiservice Edge Routers 553

Copyright © 2019, Juniper Networks, Inc.

Table 59: CoS Features and Limitations on M Series and T Series Routers 554
Table 60: Drop Priority Classification for Packet Sent from Enhanced III to Enhanced II FPC on M320 Routers 561
Table 61: Drop Priority Classification for Packet Sent from Enhanced II FPC Without Tricolor Marking to Enhanced III FPC on M320 Routers 561
Table 62: Drop Priority Classification for Packet Sent from Enhanced II FPC with Tricolor Marking to Enhanced III FPC on M320 Routers 561

Chapter 15 **Configuring Class of Service on PTX Series Packet Transport Routers . . 599**

Table 63: CoS Features and Limitations on PTX Series Routers 599
Table 64: Strict-Priority and Scheduling Processes on PTX Series Routers 606

Chapter 16 **Configuring Class of Service on T Series Core Routers 649**

Table 65: CoS Features and Limitations on M Series and T Series Routers 650

Part 4 Configuring Line Card-Specific and Interface-Specific Functionality

Chapter 17 **Feature Support of Line Cards and Interfaces 681**

Table 66: CoS Features of the Router Hardware and Interface Families Compared 681
Table 67: Scheduling on the Router Hardware and Interface Families Compared 682
Table 68: Schedulers on the Router Hardware and Interface Families Compared 682
Table 69: Queue Parameters on the Router Hardware and Interface Families Compared 683

Chapter 20 **Configuring Class of Service on IQ and Enhanced IQ (IQE) PICs 721**

Table 70: Basic Example of Excess Bandwidth 723
Table 71: Hardware Use of Basic Example Parameters 723
Table 72: Default Mode Example for IQE PICs 725
Table 73: Undersubscribed PIR Mode Example for IQE PICs 726
Table 74: Oversubscribed PIR Mode Example for IQE PICs 726
Table 75: CIR Mode Example for IQE PICs 727
Table 76: Excess Rate Mode Example for IQE PICs 728
Table 77: Default Queue Rates on the IQE PIC 728
Table 78: PIR Mode with No Excess Configuration 730
Table 79: PIR Mode with No Excess Hardware Behavior 731
Table 80: PIR Mode with Transmit Rate Configuration 731
Table 81: PIR Mode with Transmit Rate Hardware Behavior 731
Table 82: Second PIR Mode with Transmit Rate Configuration Example 732
Table 83: Second PIR Mode with Transmit Rate Hardware Behavior Example . . 732
Table 84: PIR Mode with Transmit Rate and Excess Rate Configuration 733
Table 85: PIR Mode with Transmit Rate and Excess Rate Hardware Behavior . . 733
Table 86: Excess Rate Configuration 733
Table 87: Excess Rate Hardware Behavior 734
Table 88: PIR Mode Generating Error Condition 734
Table 89: PIR Mode Generating Error Condition Behavior 734
Table 90: CIR Mode with No Excess Rate Configuration 735

Copyright © 2019, Juniper Networks, Inc.

Table 91: CIR Mode with No Excess Rate Hardware Behavior 735
Table 92: CIR Mode with Some Shaping Rates and No Excess Rate Configuration 736
Table 93: CIR Mode with Some Shaping Rates and No Excess Rate Hardware Behavior 736
Table 94: CIR Mode with Shaping Rates and Transmit Rates and No Excess Rate Configuration 737
Table 95: CIR Mode with Shaping Rates and Transmit Rates and No Excess Rate Hardware Behavior 737
Table 96: CIR Mode with Shaping Rates Greater Than Logical Interface Shaping Rate Configuration 738
Table 97: CIR Mode with Shaping Rates Greater Than Logical Interface Shaping Rate Hardware Behavior 738
Table 98: CIR Mode with Excess Rate Configuration 739
Table 99: CIR Mode with Excess Rate Hardware Behavior 739
Table 100: Oversubscribed PIR Mode with Transmit Rate Configuration 740
Table 101: Oversubscribed PIR Mode with Transmit Rate Hardware Behavior . . 740
Table 102: Oversubscribed PIR Mode with Transmit Rate and Excess Rate Configuration 741
Table 103: Oversubscribed PIR Mode with Transmit Rate and Excess Rate Hardware Behavior 741
Table 104: CIR Mode with Transmit Rate and Excess Rate Configuration 742
Table 105: CIR Mode with Transmit Rate and Excess Rate Hardware Behavior 742
Table 106: Excess Priority Configuration 743
Table 107: Default Handling of Excess Traffic 754
Table 108: Shaping Rate and WRR Calculations by PIC Type 763

Chapter 21 **Configuring Class of Service on Ethernet IQ2 and Enhanced IQ2 PICs . . . 793**

Table 109: Shaper Accuracy of 1-Gbps Ethernet at the Logical Interface Level . . 798
Table 110: Shaper Accuracy of 10-Gbps Ethernet at the Logical Interface Level 799
Table 111: Shaper Accuracy of 1-Gbps Ethernet at the Interface Set Level 799
Table 112: Shaper Accuracy of 10-Gbps Ethernet at the Interface Set Level 799
Table 113: Shaper Accuracy of 1-Gbps Ethernet at the Physical Port Level 800
Table 114: Shaper Accuracy of 10-Gbps Ethernet at the Physical Port Level . . . 800
Table 115: RTT Delay Buffers for IQ2 PICs 800
Table 116: Scheduler Allocation for an Ethernet IQ2 PIC 804

Chapter 23 **Configuring Class of Service on SONET/SDH OC48/STM16 IQE PICs . . . 861**

Table 117: Computation of CIR and PIR on the Logical Interfaces 870
Table 118: Computation of Rate for the Rate Limit Configured on the Queue with Transmit Rate Percentage 871
Table 119: Junos OS Priorities Mapped to SONET/SDH OC48/STM16 IQE PIC Hardware Priorities 873
Table 120: Queue-Level Mapping for Excess Priority and Excess Rate 874
Table 121: Priority Mapping and Output Calculation for Different Queues on the SONET/SDH OC48/STM16 IQE PIC 876
Table 122: Scaling for SONET/SDH OC48/STM16 IQE PIC 877
Table 123: Rebasing Transmission Rates and Remainder Calculation 880

Copyright © 2019, Juniper Networks, Inc.

Chapter 24 Configuring Class of Service on 10-Gigabit Ethernet LAN/WAN PICs with SFP+ . . . 897

Table 124: CoS Statements Supported on the 10-Gigabit Ethernet LAN/WAN PICs . . . 898
Table 125: Port Groups on 10-Gigabit Ethernet LAN/WAN PICs . . . 905

Chapter 25 Configuring Class of Service on Enhanced Queuing DPCs . . . 907

Table 126: IQ2 PIC and Enhanced Queuing DPC Compared . . . 908
Table 127: Junos OS Priorities Mapped to Enhanced Queuing DPC Hardware Priorities . . . 914
Table 128: Shaping Rates and WFQ Weights . . . 916
Table 129: Sample Shaping Rates and WFQ Weights . . . 917
Table 130: Rounding Configured Weights to Hardware Weights . . . 917
Table 131: Allocating Weights with PIR and CIR on Logical Interfaces . . . 918
Table 132: Sharing Bandwidth Among Logical Interfaces . . . 919
Table 133: First Example of Bandwidth Sharing . . . 919
Table 134: Second Example of Bandwidth Sharing . . . 920
Table 135: Final Example of Bandwidth Sharing . . . 920

Chapter 26 Configuring Class of Service on MICs, MPCs, and MLCs . . . 929

Table 136: CoS Limitations on MIC and MPC Interfaces . . . 930
Table 137: Dedicated Queues for MPCs . . . 932
Table 138: MPC and MIC support for per-VLAN (logical interface) queuing . . . 935
Table 139: Number of VLANs on 16x10G, MPC3E, MPC4E and MPC6E . . . 937
Table 140: Number of VLANs on MPC3E-NG/MPC2E-NG, MPC5E . . . 939
Table 141: MPCs and the Junos OS Release that Support Flexible Queuing . . . 942
Table 142: Shaping Rate Granularity for MPC Ports . . . 954
Table 143: Shaping Priority and Default Excess Shaping Priority . . . 956
Table 144: Applying Traffic Control Profiles . . . 966
Table 145: Default Maximum Queues and Corresponding Rate Wheel Update Periods . . . 992
Table 146: Default Shaping Granularities on Non-Enhanced Queuing MICs and MPCs . . . 993
Table 147: Hardware Requirements for L2TP LNS Inline Services . . . 1002

Part 5 Configuration Statements and Operational Commands

Chapter 29 Operational Commands . . . 1327

Table 148: show class-of-service classifier Output Fields . . . 1328
Table 149: show class-of-service code-point-aliases Output Fields . . . 1331
Table 150: show class-of-service drop-profile Output Fields . . . 1333
Table 151: show class-of-service fabric scheduler-map Output Fields . . . 1337
Table 152: show class-of-service fabric statistics Output Fields . . . 1340
Table 153: show class-of-service forwarding-table classifier Output Fields . . . 1346
Table 154: show class-of-service forwarding-table classifier mapping Output Fields . . . 1348
Table 155: show class-of-service forwarding-table drop-profile Output Fields . . . 1350
Table 156: show class-of-service forwarding-table fabric scheduler-map Output Fields . . . 1352

Copyright © 2019, Juniper Networks, Inc.

Table 157: show class-of-service forwarding-table rewrite-rule Output Fields . . 1354
Table 158: show class-of-service forwarding-table rewrite-rule mapping Output Fields 1356
Table 159: show class-of-service forwarding-table scheduler-map Output Fields 1358
Table 160: show class-of-service forwarding-table traffic-class-map Output Fields 1360
Table 161: show class-of-service forwarding-table traffic-class-map mapping Output Fields 1361
Table 162: show class-of-service fragmentation-map Output Fields 1362
Table 163: show class-of-service interface Output Fields 1365
Table 164: show class-of-service loss-priority-rewrite Output Fields 1395
Table 165: show class-of-service l2tp-session Output Fields 1397
Table 166: show class-of-service policy-map Output Fields 1399
Table 167: show class-of-service rewrite-rule Output Fields 1401
Table 168: show class-of-service routing-instance Output Fields 1403
Table 169: show class-of-service scheduler-hierarchy interface Output Fields 1404
Table 170: show class-of-service scheduler-map Output Fields 1406
Table 171: show class-of-service traffic-class-map Output Fields 1410
Table 172: show class-of-service translation-table Output Fields 1411
Table 173: show interfaces forwarding-class-counters Output Fields 1417
Table 174: Layer 2 Overhead and Transmitted Packets or Byte Counts 1422
Table 175: show interfaces queue Output Fields 1426
Table 176: Byte Count by Interface Hardware 1429
Table 177: show interfaces voq Output Fields 1469

Copyright © 2019, Juniper Networks, Inc.

About the Documentation

- Documentation and Release Notes on page xxxvii
- Using the Examples in This Manual on page xxxvii
- Documentation Conventions on page xxxix
- Documentation Feedback on page xli
- Requesting Technical Support on page xli

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 https://www.juniper.net/documentation/.

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 https://www.juniper.net/books.

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.

Copyright © 2019, Juniper Networks, Inc.

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; }
   ```


Copyright © 2019, Juniper Networks, Inc.

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 CLI Explorer.

Documentation Conventions

Table 1 on page xxxix 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 xl defines the text and syntax conventions used in this guide.

Copyright © 2019, Juniper Networks, Inc.

Table 2: Text and Syntax Conventions

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

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

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

GUI Conventions

Copyright © 2019, Juniper Networks, Inc.

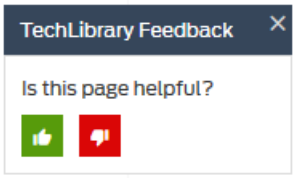
Table 2: Text and Syntax Conventions (continued)

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

Documentation Feedback

We encourage you to provide feedback so that we can improve our documentation. You can use either of the following methods:

- Online feedback system—Click TechLibrary Feedback, on the lower right of any page on the Juniper Networks TechLibrary site, and do one of the following:



  - Click the thumbs-up icon if the information on the page was helpful to you.
  - Click the thumbs-down icon if the information on the page was not helpful to you or if you have suggestions for improvement, and use the pop-up form to provide feedback.

- E-mail—Send your comments to **techpubs-comments@juniper.net**. Include the document or topic name, URL or page number, and software 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 Juniper Care or Partner Support Services 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 https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit https://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.

Copyright © 2019, Juniper Networks, Inc.

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: https://www.juniper.net/customers/support/
- Search for known bugs: https://prsearch.juniper.net/
- Find product documentation: https://www.juniper.net/documentation/
- Find solutions and answer questions using our Knowledge Base: https://kb.juniper.net/
- Download the latest versions of software and review release notes: https://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: https://www.juniper.net/company/communities/
- Create a service request online: https://myjuniper.juniper.net

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

Creating a Service Request with JTAC

You can create a service request with JTAC on the Web or by telephone.

- Visit https://myjuniper.juniper.net.
- 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 https://support.juniper.net/support/requesting-support/.

Copyright © 2019, Juniper Networks, Inc.

PART 1

Overview

- Understanding How Class of Service Manages Congestion and Defines Traffic Forwarding Behavior on page 3

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 1

Understanding How Class of Service Manages Congestion and Defines Traffic Forwarding Behavior

- Understanding How Class of Service Manages Congestion and Controls Service Levels in the Network on page 4
- How CoS Applies to Packet Flow Across a Network on page 6
- The Junos OS CoS Components Used to Manage Congestion and Control Service Levels on page 7
- Mapping CoS Component Inputs to Outputs on page 11
- Default Junos OS CoS Settings on page 14
- Packet Flow Through the Junos OS CoS Process Overview on page 17
- Configuring Basic Packet Flow Through the Junos OS CoS Process on page 19
- Example: Classifying All Traffic from a Remote Device by Configuring Fixed Interface-Based Classification on page 25
- Interface Types That Do Not Support Junos OS CoS on page 31

Copyright © 2019, Juniper Networks, Inc.

Understanding How Class of Service Manages Congestion and Controls Service Levels in the Network

Usually, IP routers forward packets independently and without any control on throughput or delay. This is known as *best-effort* service. This service is as good as the network equipment and links, and the result is satisfactory for many traditional IP applications emphasizing data delivery, such as e-mail or Web browsing. However, IP applications such as real-time video and audio (or voice) require lower delay, jitter, and loss parameters than simple best-effort networks can provide during times of network congestion.

When a network experiences congestion and delay, some packets must be dropped. The Juniper Networks Junos operating system (Junos OS) class of service (CoS) enables you to assign traffic to classes and offer various levels of throughput and packet loss when congestion occurs.

CoS is the assignment of traffic flows to different service levels. Service providers can use router-based CoS features to define service levels that provide different delay, jitter (delay variation), and packet loss characteristics to particular applications served by specific traffic flows.

A router cannot compromise best-effort forwarding performance in order to deliver CoS features, because this merely trades one problem for another. When CoS features are enabled, they must allow routers to better process critical packets as well as best-effort traffic flows, even during times of congestion. Network throughput is determined by a combination of available bandwidth and delay. CoS guarantees a minimum bandwidth dedicated to a service class.

The main impact of CoS on network delay is in queuing delays, when packets are normally queued for output in the order of arrival, regardless of service class. Queuing delays increase with network congestion and often result in lost packets when queue buffers overflow. The other two elements of overall network delay, serial transmission delays determined by link speeds and propagation delays determined by media type, are not determined by CoS settings.

For interfaces that carry IPv4, IPv6, and MPLS traffic, you can configure the Junos OS CoS features to provide multiple classes of service for different applications. On the routing device, you can configure multiple forwarding classes for transmitting packets, define which packets are placed into each output queue, schedule the transmission service level for each queue, and manage congestion using a random early detection (RED) algorithm.

The Junos OS CoS features provide a set of mechanisms that you can use to provide differentiated services when best-effort traffic delivery is insufficient. In designing CoS applications, you must give careful consideration to your service needs, and you must thoroughly plan and design your CoS configuration to ensure consistency across all routing devices in a CoS domain. You must also consider all the routing devices and other networking equipment in the CoS domain to ensure interoperability among all equipment.

Copyright © 2019, Juniper Networks, Inc.

CoS Applications

You can configure CoS features to meet the needs of multiple applications. Because the components are generic, you can use a single CoS configuration syntax across multiple routing devices. CoS mechanisms are useful for two broad classes of applications. These applications can be referred to as *in the box* and *across the network*.

In-the-box applications use CoS mechanisms to provide special treatment for packets passing through a single node on the network. You can monitor the incoming traffic on each interface, using CoS to provide preferred service to some interfaces (that is, to some customers) while limiting the service provided to other interfaces. You can also filter outgoing traffic by the packet's destination, thus providing preferred service to some destinations.

Across-the-network applications use CoS mechanisms to provide differentiated treatment to different classes of packets across a set of nodes in a network. In these types of applications, you typically control the ingress and egress routing devices to a routing domain and all the routing devices within the domain. You can use the Junos OS CoS features to modify packets traveling through the domain to indicate the packet's priority across the domain.

Specifically, you modify the CoS code points in packet headers, remapping these bits to values that correspond to levels of service. When all routing devices in the domain are configured to associate the precedence bits with specific service levels, packets with the same code points traveling across the domain receive the same level of service from the ingress point to the egress point. For CoS to work in this case, the mapping between the code points and service levels must be identical across all routing devices in the domain.

The Junos OS CoS applications support the following range of mechanisms:

- Differentiated Services (DiffServ)—The CoS application supports DiffServ, which uses a 6-bit differentiated services code point (DSCP) in the differentiated services field of the IPv4 and IPv6 packet header. For IPv6, DSCP is referred to as traffic class. The configuration uses DSCP values to determine the forwarding class associated with each packet. IPv4 traffic can also use the 3-bit IP precedence bits to classify traffic.
- Layer 2 to Layer 3 CoS mapping—The CoS application supports mapping of Layer 2 (IEEE 802.1p) packet headers to routing device forwarding class and loss-priority values.

  Layer 2 to Layer 3 CoS mapping involves setting the forwarding class and loss priority based on information in the Layer 2 header. Output involves mapping the forwarding class and loss priority to a Layer 2-specific marking. You can mark the Layer 2 and Layer 3 headers simultaneously.
- MPLS EXP—Supports configuration of mapping of MPLS experimental (EXP) bit settings to routing device forwarding classes and vice versa.
- VPN outer-label marking—Supports setting of outer-label EXP bits, also known as CoS bits, based on MPLS EXP mapping.

Copyright © 2019, Juniper Networks, Inc.

CoS Standards

The standards for Junos OS class of service (CoS) capabilities are defined in the following RFCs:

- RFC 2474, *Definition of the Differentiated Services Field in the IPv4 and IPv6 Headers*
- RFC 2597, *Assured Forwarding PHB Group*
- RFC 2598, *An Expedited Forwarding PHB*
- RFC 2698, *A Two Rate Three Color Marker*

Related Documentation

- The Junos OS CoS Components Used to Manage Congestion and Control Service Levels on page 7

How CoS Applies to Packet Flow Across a Network

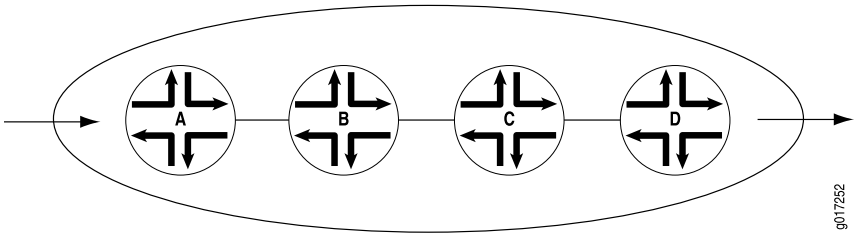
CoS works by examining traffic entering at the edge of your network. The edge routing devices classify traffic into defined service groups to provide the special treatment of traffic across the network. For example, voice traffic can be sent across certain links, and data traffic can use other links. In addition, the data traffic streams can be serviced differently along the network path to ensure that higher-paying customers receive better service. As the traffic leaves the network at the far edge, you can reclassify the traffic.

To support CoS, you must configure each routing device in the network. Generally, each routing device examines the packets that enter it to determine their CoS settings. These settings then dictate which packets are first transmitted to the next downstream routing device. In addition, the routing devices at the edges of the network might be required to alter the CoS settings of the packets that enter the network from the customer or peer networks.

In Figure 1 on page 7, Router A is receiving traffic from a customer network. As each packet enters, Router A examines the packet's current CoS settings and classifies the traffic into one of the groupings defined by the Internet service provider (ISP). This definition allows Router A to prioritize its resources for servicing the traffic streams it is receiving. In addition, Router A might alter the CoS settings (forwarding class and loss priority) of the packets to better match the ISP's traffic groups. When Router B receives the packets, it examines the CoS settings, determines the appropriate traffic group, and processes the packet according to those settings. It then transmits the packets to Router C, which performs the same actions. Router D also examines the packets and determines the appropriate group. Because Router D sits at the far end of the network, the ISP might decide once again to alter the CoS settings of the packets before Router D transmits them to the neighboring network.

Copyright © 2019, Juniper Networks, Inc.

Figure 1: Packet Flow Across the Network



Related Documentation

- The Junos OS CoS Components Used to Manage Congestion and Control Service Levels on page 7

The Junos OS CoS Components Used to Manage Congestion and Control Service Levels

Any CoS implementation must work consistently end to end through the network. A standards-based, vendor-neutral CoS implementation satisfies this requirement best. Junos OS CoS features interoperate with other vendors' CoS implementations because they are based on IETF Differentiated Services (DiffServ) standards. Junos OS CoS consists of many components that you can combine and tune to provide the level of services required by customers.

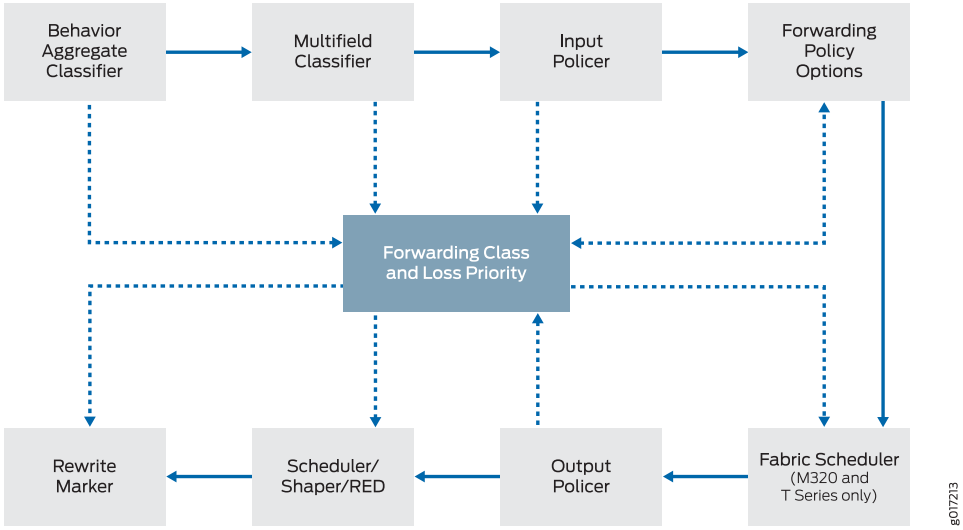
DiffServ specifications establish a six-bit field in the IPv4 and IPv6 packet header to indicate the service class that should be applied to the packet. The bit values in the DiffServ field form DiffServ code points (DSCPs) that can be set by the application or a router on the edge of a DiffServ-enabled network.

Although CoS methods such as DiffServ specify the position and length of the DSCP in the packet header, the implementation of the router mechanisms to deliver DiffServ internally is vendor-specific. CoS functions in Junos OS are configured through a series of mechanisms that you can configure individually or in combination to define particular service offerings.

Figure 2 on page 8 shows the components of the Junos OS CoS features, illustrating the sequence in which they interact.

Copyright © 2019, Juniper Networks, Inc.

Figure 2: Packet Flow Through CoS-Configurable Components



You can configure one or more of the following Junos OS CoS mechanisms:

- Classifiers—*Packet classification* refers to the examination of an incoming packet. This function associates the packet with a particular CoS servicing level. In Junos OS, classifiers associate incoming packets with a forwarding class and loss priority and, based on the associated forwarding class, assign packets to output queues. Two general types of classifiers are supported:
  - Behavior aggregate classifiers—A *behavior aggregate* (BA) is a method of classification that operates on a packet as it enters the routing device. The CoS value in the packet header is examined, and this single field determines the CoS settings applied to the packet. BA classifiers allow you to set the forwarding class and loss priority of a packet based on the Differentiated Services code point (DSCP) value, DSCP IPv6 value, IP precedence value, MPLS EXP bits, and IEEE 802.1p value. The default classifier is based on the IP precedence value.

    (You can also configure *code-point aliases* which assign a name to a pattern of code-point bits. You can use this name instead of the bit pattern when you configure other CoS components, such as classifiers, drop-profile maps, and rewrite rules.)

    See "Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic" on page 36 for more information on BA classifiers.
  - Multifield traffic classifiers—A *multifield* classifier is a second method for classifying traffic flows. Unlike a behavior aggregate, a multifield classifier can examine multiple fields in the packet. Examples of some fields that a multifield classifier can examine include the source and destination address of the packet as well as the source and destination port numbers of the packet. With multifield classifiers, you set the forwarding class and loss priority of a packet based on firewall filter rules. Multifield classification is usually done at the edge of the network for packets that do not have valid or trusted behavior aggregate code points.

Copyright © 2019, Juniper Networks, Inc.

See "Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields" on page 97 for more information on multifield classifiers.

- Forwarding classes—The *forwarding classes* affect the forwarding, scheduling, and marking policies applied to packets as they transit a routing device. Known as ordered aggregates in the DiffServ architecture, the forwarding class plus the loss priority determine the router's per-hop behavior (PHB in DiffServ) for CoS. Four categories of forwarding classes are supported: best effort, assured forwarding, expedited forwarding, and network control. For most Juniper Networks M Series Multiservice Edge Routers, four forwarding classes are supported. You can configure up to one each of the four types of forwarding classes. For M120 and M320 Multiservice Edge Routers, Juniper Networks MX Series 5G Universal Routing Platforms, Juniper Networks T Series Core Routers, and EX Series switches, 16 forwarding classes are supported, so you can classify packets more granularly. For example, you can configure multiple classes of expedited forwarding (EF) traffic: EF, EF1, and EF2.

  See "Understanding How Forwarding Classes Assign Classes to Output Queues" on page 205 for more information on forwarding classes.

- Loss priorities—*Loss priorities* allow you to set the priority of dropping a packet. Loss priority affects the scheduling of a packet without affecting the packet's relative ordering. You can use the packet loss priority (PLP) bit as part of a congestion control strategy. You can use the loss priority setting to identify packets that have experienced congestion. Typically you mark packets exceeding some service level with a high loss priority. You set loss priority by configuring a classifier or a policer. The loss priority is used later in the workflow to select one of the drop profiles used by RED.

  See "Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows" on page 361 for more information on packet loss priorities.

- Forwarding policy options—These options allow you to associate forwarding classes with next hops. Forwarding policy options also allow you to create classification overrides, which assign forwarding classes to sets of prefixes.

  See "Forwarding Policy Options Overview" on page 223 for more information on forwarding policy options.

- Transmission scheduling and rate control—These parameters provide you with a variety of tools to manage traffic flows:

  - Queuing—After a packet is sent to the outgoing interface on a routing device, it is queued for transmission on the physical media. The amount of time a packet is queued on the routing device is determined by the availability of the outgoing physical media as well as the amount of traffic using the interface.

  - Schedulers—An individual routing device interface has multiple queues assigned to store packets. The routing device determines which queue to service based on a particular method of scheduling. This process often involves a determination of which type of packet should be transmitted before another. The Junos OS schedulers allow you to define the priority, bandwidth, delay buffer size, rate control status, and RED drop profiles to be applied to a particular queue for packet transmission.

    See "How Schedulers Define Output Queue Properties" on page 254 for more information on schedulers.

Copyright © 2019, Juniper Networks, Inc.

- Fabric schedulers—For M120, M320, and T Series routers only, fabric schedulers allow you to identify a packet as high or low priority based on its forwarding class, and to associate schedulers with the fabric priorities.
- Policers for traffic classes—*Policers* allow you to limit traffic of a certain class to a specified bandwidth and burst size. Packets exceeding the policer limits can be discarded (hard policing), or can be assigned to a different forwarding class, a different loss priority, or both (soft policing). You define policers with filters that can be associated with input or output interfaces.

  See "Controlling Network Access Using Traffic Policing Overview" on page 115 for more information on policers.

- Rewrite rules—A *rewrite rule* sets the appropriate CoS bits in the outgoing packet. This allows the next downstream routing device to classify the packet into the appropriate service group. Rewriting, or marking, outbound packets is useful when the routing device is at the border of a network and must alter the CoS values to meet the policies of the targeted peer.

  Typically, rewrites of the DSCPs on outgoing packets are done once, when packets enter the DiffServ portion of the network, either because the packets do not arrive from the customer with the proper DSCP bit set or because the service provider wants to verify that the customer has set the DSCP properly. CoS schemes that accept the DSCP and classify and schedule traffic solely on DSCP value perform behavior aggregate (BA) DiffServ functions and do not usually rewrite the DSCP. DSCP rewrites typically occur in multifield (MF) DiffServ scenarios.

  See "Rewriting Packet Headers to Ensure Forwarding Behavior" on page 388 for more information on rewrite rules.

Related Documentation

- Understanding How Class of Service Manages Congestion and Controls Service Levels in the Network on page 4

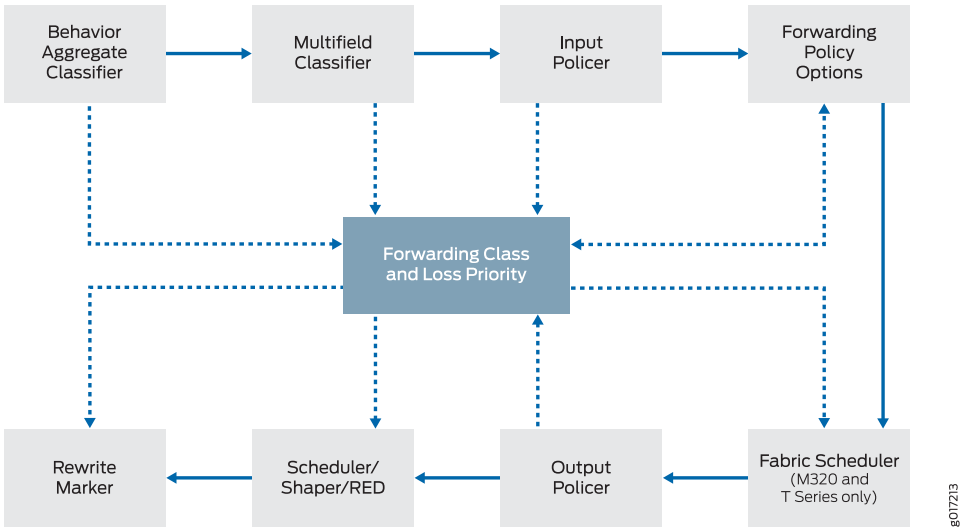
Copyright © 2019, Juniper Networks, Inc.

Mapping CoS Component Inputs to Outputs

Some CoS components map one set of values to another set of values. Each mapping contains one or more inputs and one or more outputs.

Figure 2 on page 8 shows the components of the Junos OS CoS features, illustrating the sequence in which they interact.

Figure 3: Packet Flow Through CoS-Configurable Components



TIP: Component mapping enables you to define forwarding classes and packet loss priorities for various traffic flows and then map those forwarding classes to output queues with specific shaping and scheduling characteristics.

When you configure a mapping, you set the outputs for a given set of inputs, as shown in Table 3 on page 11.

Table 3: CoS Mappings—Inputs and Outputs

CoS Mappings	Inputs	Outputs	Comments
classifiers	code-points	forwarding-class loss-priority	The map sets the forwarding class and PLP for a specific set of code points.
drop-profile-map	loss-priority protocol	drop-profile	The map sets the drop profile for a specific PLP and protocol type.
scheduler-maps	forwarding-class	scheduler	This map assigns a forwarding class to a specific scheduler.

Copyright © 2019, Juniper Networks, Inc.

Table 3: CoS Mappings—Inputs and Outputs (continued)

CoS Mappings	Inputs	Outputs	Comments
rewrite-rules	forwarding-class loss-priority	code-points	The map sets the code points for a specific forwarding class and PLP.

Following are sample configurations for classifiers, drop-profile maps, scheduler maps, and rewrite rules.

Copyright © 2019, Juniper Networks, Inc.

In the following classifier sample configuration, packets with EXP bits **000** are assigned to the **data-queue** forwarding class with a **low** loss priority, and packets with EXP bits **001** are assigned to the **data-queue** forwarding class with a **high** loss priority.

```
[edit class-of-service]
classifiers {
  exp exp_classifier {
    forwarding-class data-queue {
      loss-priority low code-points 000;
      loss-priority high code-points 001;
    }
  }
}
```

See "Configuring Behavior Aggregate Classifiers" on page 51 for more information on setting the forwarding class and loss priority for a specific set of code-point aliases and bit patterns

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

```
[edit class-of-service]
schedulers {
  best-effort {
    drop-profile-map loss-priority low protocol any drop-profile low-drop;
    drop-profile-map loss-priority high protocol any drop-profile high-drop;
  }
}
```

See "Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers" on page 359 for more information on mapping drop profiles to a scheduler.

In the following scheduler maps configuration sample, each of the default forwarding classes is mapped to a scheduler specifically designed for that forwarding class.

```
scheduler-maps {
    basic {
        forwarding-class best-effort scheduler be;
        forwarding-class assured-forwarding scheduler af;
        forwarding-class expedited-forwarding scheduler ef;
        forwarding-class network-control scheduler nc;
    }
}
```

See "Configuring Scheduler Maps" on page 258 for more information on mapping forwarding classes to schedulers.

Copyright © 2019, Juniper Networks, Inc.

In the following rewrite rule configuration sample, packets in the **be** forwarding class with **low** loss priority are assigned the EXP bits **000**, and packets in the **be** forwarding class with **high** loss priority are assigned the EXP bits **001**.

```
[edit class-of-service]
rewrite-rules {
   exp exp-rw {
      forwarding-class be {
         loss-priority low code-point 000;
         loss-priority high code-point 001;
      }
   }
}
```

See "Configuring Rewrite Rules" on page 391 for more information on setting the code-point aliases and bit patterns for specific forwarding classes and loss priorities as packets leave the device.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 359
- Configuring Scheduler Maps on page 258
- Applying Default Rewrite Rules on page 389

Default Junos OS CoS Settings

If you do not configure any CoS settings on your router, the software performs some CoS functions to ensure that user traffic and protocol packets are forwarded with minimum delay when the network is experiencing congestion. Some default mappings are automatically applied to each logical interface that you configure. Other default mappings, such as explicit default classifiers and rewrite rules, are in operation only if you explicitly associate them with an interface.

You can display default CoS settings by issuing the **show class-of-service** operational mode command. This section includes sample output displaying the default CoS settings. The sample output is truncated for brevity.

show class-of-service

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

NOTE: Some platforms require an argument after the show class-of-service command. The argument is to select a portion of the following output to display.

Copyright © 2019, Juniper Networks, Inc.

Default Forwarding Classes

```
Forwarding class           Queue
  best-effort                      0
  expedited-forwarding             1
  assured-forwarding               2
  network-control                  3
```

Default Code-Point Aliases

```
Code point type: dscp
  Alias              Bit pattern
  af11               001010
  af12               001100
...
Code point type: dscp-ipv6
...
Code point type: exp
...
Code point type: ieee-802.1
...
Code point type: inet-precedence
...
Code point type: ieee-802.1ad
...
```

Default Classifiers

```
Classifier: dscp-default, Code point type: dscp, Index: 7
...

Classifier: dscp-ipv6-default, Code point type: dscp-ipv6, Index: 8
...

Classifier: exp-default, Code point type: exp, Index: 9
...

Classifier: ieee8021p-default, Code point type: ieee-802.1, Index: 10
...

Classifier: ipprec-default, Code point type: inet-precedence, Index: 11
...

Classifier: ipprec-compatibility, Code point type: inet-precedence, Index: 12
...

Classifier: ieee8021ad-default, Code point type: ieee-802.1ad, Index: 41
...
```

Default Frame Relay Loss Priority Map

```
Loss-priority-map: frame-relay-de-default, Code point type: frame-relay-de, Index:
 13
  Code point         Loss priority
  0                  low
  1                  high
```

Copyright © 2019, Juniper Networks, Inc.

Default Rewrite Rules

```
Rewrite rule: dscp-default, Code point type: dscp, Index: 24
  Forwarding class                  Loss priority       Code point
  best-effort                       low                 000000
  best-effort                       high                000000
...

Rewrite rule: dscp-ipv6-default, Code point type: dscp-ipv6, Index: 25
...

Rewrite rule: exp-default, Code point type: exp, Index: 26
...

Rewrite rule: ieee8021p-default, Code point type: ieee-802.1, Index: 27
...

Rewrite rule: ipprec-default, Code point type: inet-precedence, Index: 28
...

Rewrite rule: ieee8021ad-default, Code point type: ieee-802.1ad, Index: 42
...
```

Default Drop Profile

```
Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
```

Default Schedulers

```
Scheduler map: <default>, Index: 2

  Scheduler: <default-be>, Forwarding class: best-effort, Index: 17
    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>
       High            Any             1    <default-drop-profile>
...
```

Related Documentation

- Default Forwarding Classes on page 208
- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351
- Default Schedulers Overview on page 257
- Forwarding Classes and Fabric Priority Queues on page 234

Copyright © 2019, Juniper Networks, Inc.

Packet Flow Through the Junos OS CoS Process Overview

Perhaps the best way to understand Junos OS CoS is to examine how a packet is treated on its way through the CoS process. This topic includes a description of each step and figures illustrating the process.

The following steps describe the CoS process:

1. A logical interface has one or more classifiers of different types applied to it (at the **[edit class-of-service interfaces]** hierarchy level). The types of classifiers are based on which part of the incoming packet the classifier examines (for example, EXP bits, IEEE 802.1p bits, or DSCP bits). You can use a translation table to rewrite the values of these bits on ingress.

   NOTE: You can only rewrite the values of these bits on ingress on the Juniper Networks M40e, M120, M320 Multiservice Edge Routers, and T Series Core Routers with IQE PICs. For more information about rewriting the values of these bits on ingress, see "Configuring ToS Translation Tables" on page 745.

2. The classifier assigns the packet to a forwarding class and a loss priority (at the **[edit class-of-service classifiers]** hierarchy level).
3. Each forwarding class is assigned to a queue (at the **[edit class-of-service forwarding-classes]** hierarchy level).
4. Input (and output) policers meter traffic and might change the forwarding class and loss priority if a traffic flow exceeds its service level.
5. The physical or logical interface has a scheduler map applied to it (at the **[edit class-of-service interfaces]** hierarchy level).

   At the **[edit class-of-service interfaces]** hierarchy level, the **scheduler-map** and **rewrite-rules** statements affect the outgoing packets, and the **classifiers** statement affects the incoming packets.
6. The scheduler defines how traffic is treated in the output queue—for example, the transmit rate, buffer size, priority, and drop profile (at the **[edit class-of-service schedulers]** hierarchy level).
7. The scheduler map assigns a scheduler to each forwarding class (at the **[edit class-of-service scheduler-maps]** hierarchy level).
8. The drop-profile defines how aggressively to drop packets that are using a particular scheduler (at the **[edit class-of-service drop-profiles]** hierarchy level).
9. The rewrite rule takes effect as the packet leaves a logical interface that has a rewrite rule configured (at the **[edit class-of-service rewrite-rules]** hierarchy level). The rewrite rule writes information to the packet (for example, EXP or DSCP bits) according to the forwarding class and loss priority of the packet.

Copyright © 2019, Juniper Networks, Inc.

Figure 4 on page 18 and Figure 5 on page 18 show the components of the Junos OS CoS features, illustrating the sequence in which they interact.

Figure 4: CoS Classifier, Queues, and Scheduler

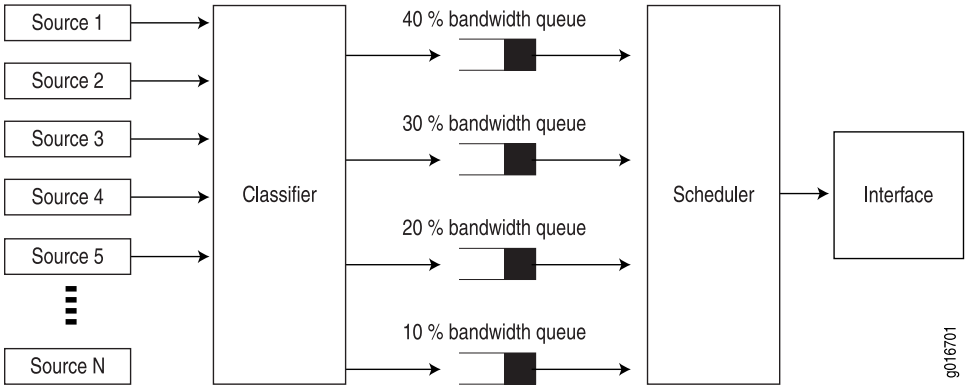
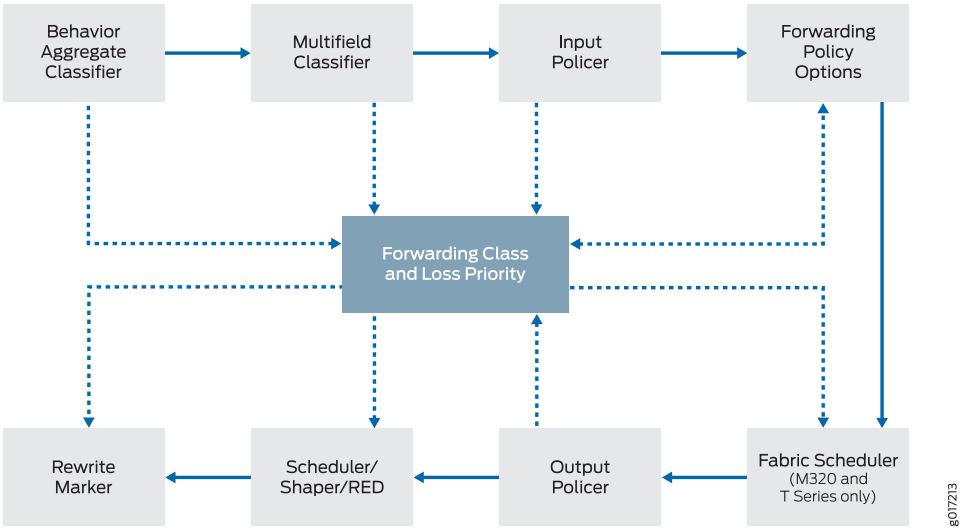


Figure 5: Packet Flow Through CoS- Configurable Components



Each outer box in Figure 5 on page 18 represents a process component. The components in the upper row apply to inbound packets, and the components in the lower row apply to outbound packets. The arrows with the solid lines point in the direction of packet flow.

The middle box (forwarding class and loss priority) represents two data values that can either be inputs to or outputs of the process components. The arrows with the dotted lines indicate inputs and outputs (or settings and actions based on settings). For example, the multifield classifier sets the forwarding class and loss priority of incoming packets. This means that the forwarding class and loss priority are outputs of the classifier; thus, the arrow points away from the classifier. The scheduler receives the forwarding class and loss priority settings, and queues the outgoing packet based on those settings. This means that the forwarding class and loss priority are inputs to the scheduler; thus, the arrow points to the scheduler.

Copyright © 2019, Juniper Networks, Inc.

Typically, only a combination of some components (not all) is used to define a CoS service offering.

Packet Flow Within Routers Overview

Although the architecture of Juniper Networks routers different in detail, the overall flow of a packet within the router remains consistent.

When a packet enters a Juniper Networks router, the PIC or other interface type receiving the packet retrieves it from the network and verifies that the link-layer information is valid. The packet is then passed to the concentrator device such as a Flexible PIC Concentrator (FPC), where the data link and network layer information is verified. In addition, the FPC is responsible for segmenting the packet into 64-byte units called J-cells. These cells are then written into packet storage memory while a notification cell is sent to the route lookup engine. The destination address listed in the notification cell is located in the forwarding table, and the next hop of the packet is written into the result cell. This result cell is queued on the appropriate outbound FPC until the outgoing interface is ready to transmit the packet. The FPC then reads the J-cells out of memory, re-forms the original packet, and sends the packet to the outgoing PIC, where it is transmitted back into the network.

Related Documentation

- Configuring Basic Packet Flow Through the Junos OS CoS Process on page 19
- Packet Flow on Juniper Networks M Series Multiservice Edge Routers on page 561
- Packet Flow on MX Series 5G Universal Routing Platforms on page 570
- Packet Flow on Juniper Networks T Series Core Routers on page 656

Configuring Basic Packet Flow Through the Junos OS CoS Process

Figure 6 on page 19 and Figure 7 on page 20 show the components of the Junos OS CoS features, illustrating the sequence in which they interact.

Figure 6: CoS Classifier, Queues, and Scheduler

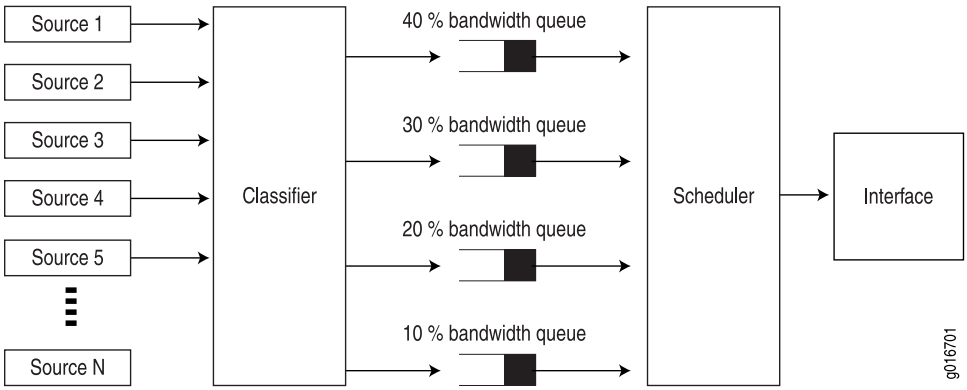
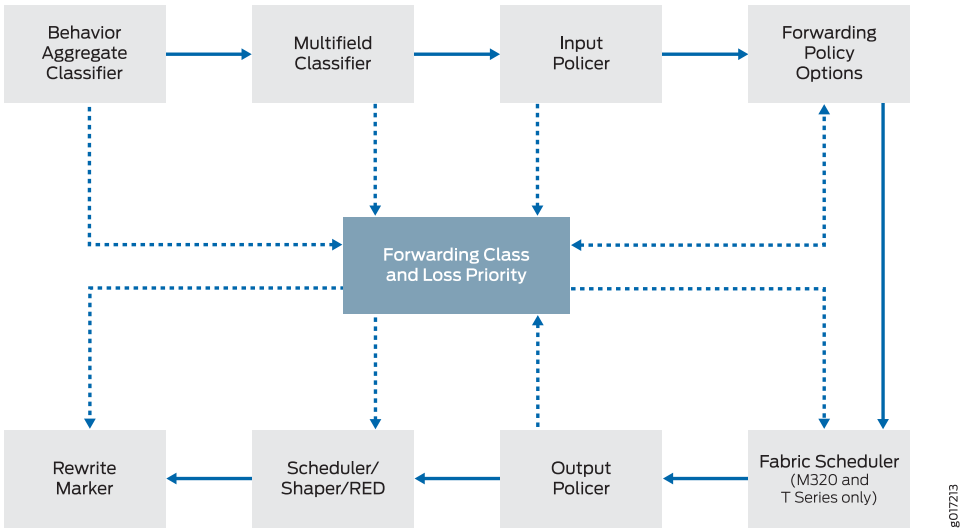


Figure 7: Packet Flow Through CoS- Configurable Components



The following configuration demonstrates the packet flow through the CoS process:

- Define Classifiers on page 20
- Apply Classifiers to Incoming Packets on Interfaces on page 21
- Define Policers to Limit Traffic and Control Congestion on page 22
- Define Drop Profiles on page 23
- Assign Each Forwarding Class to a Queue on page 23
- Define Schedulers on page 23
- Define Scheduler Maps on page 24
- Define CoS Header Rewrite Rules on page 24
- Apply Scheduler Maps and Rewrite Rules to Egress Interfaces on page 25

Define Classifiers

If you trust the CoS values in the packet headers, you can use behavior aggregate classification to map those values to a forwarding class and drop priority. For example:

```
[edit class-of-service]
classifiers {
exp exp_classifier {
   forwarding-class data-queue {
      loss-priority low code-points 000;
      loss-priority high code-points 001;
   }
   forwarding-class video-queue {
      loss-priority low code-points 010;
      loss-priority high code-points 011;
   }
   forwarding-class voice-queue {
```


Copyright © 2019, Juniper Networks, Inc.

```
      loss-priority low code-points 100;
      loss-priority high code-points 101;
    }
    forwarding-class nc-queue {
      loss-priority low code-points 110;
      loss-priority high code-points 111;
    }
}
```

If you do not trust the CoS values in the packet headers, you can use the more complex multifield classification to map ingress traffic to a forwarding class and drop priority. For example:

```
[edit firewall]
family inet {
  filter classify {
    term sip {
      from {
        protocol [ udp tcp ];
        port 5060;
      }
      then {
        forwarding-class nc-queue;
        loss-priority low;
        accept;
      }
    }
  }
}
```

See Also

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 97

Apply Classifiers to Incoming Packets on Interfaces

You apply behavior aggregate classifiers to logical interfaces at the **[edit class-of-service interfaces]** hierarchy level. For example:

```
[edit class-of-service]
interfaces {
  so-* {
    unit 0 {
      classifiers {
        exp exp_classifier;
      }
    }
  }
  t3-* {
    unit 0 {
      classifiers {
```

Copyright © 2019, Juniper Networks, Inc.

```
            exp exp_classifier;
        }
    }
  }
}
```

You apply multifield classifiers as input filters to logical interfaces at the **[edit interfaces]** hierarchy level. For example:

```
[edit interfaces]
fe-0/0/2 {
  unit 0 {
    family inet {
      filter {
        input classify;
      }
      address 10.12.0.13/30;
    }
  }
}
```

Define Policers to Limit Traffic and Control Congestion

If you need to rate-limit a traffic flow, either by discarding excess traffic (hard policing) or reassign excess traffic to a different forwarding class and/or loss priority (soft policing), define a policier and apply the policer to a firewall filter for that traffic flow. For example:

```
[edit firewall]
policer be-lp {
  if-exceeding {
    bandwidth-limit 10m;
    burst-size-limit 62500;
  }
  then loss-priority high;
}
family inet {
  filter be-lp {
    term t1 {
      from {
        protocol tcp;
        port 80;
      }
      then policer be-lp;
      then loss-priority low;
      then accept;
    }
  }
}
```

See Also

- Controlling Network Access Using Traffic Policing Overview on page 115

Copyright © 2019, Juniper Networks, Inc.

Define Drop Profiles

Use drop profiles to define the drop probabilities across the range of delay-buffer occupancy, supporting the random early detection (RED) process.

```
[edit class-of-service]
drop-profiles {
   be-red {
      fill-level 20 drop-probability 25;
      fill-level 30 drop-probability 50;
      fill-level 40 drop-probability 75;
      fill-level 50 drop-probability 100;
   }
}
```

See Also

- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351

Assign Each Forwarding Class to a Queue

To provide differentiated services to each forwarding class, assign each forwarding class to it's own output queue. For example:

```
[edit class-of-service]
forwarding-classes {
   queue 0 data-queue;
   queue 1 video-queue;
   queue 2 voice-queue;
   queue 3 nc-queue;
}
```

See Also

- Understanding How Forwarding Classes Assign Classes to Output Queues on page 205

Define Schedulers

Define the scheduler characteristics for each forwarding class. For example:

```
[edit class-of-service]
schedulers { #
   data-scheduler {
      transmit-rate percent 50;
      buffer-size percent 50;
      priority low;
      drop-profile-map loss-priority high protocol any drop-profile be-red;
   }
   video-scheduler {
      transmit-rate percent 25;
      buffer-size percent 25;
      priority strict-high;
   }
   voice-scheduler {
```

Copyright © 2019, Juniper Networks, Inc.

```
    transmit-rate percent 20;
    buffer-size percent 20;
    priority high;
  }
  nc-scheduler {
    transmit-rate percent 5;
    buffer-size percent 5;
    priority high;
  }
}
```

See Also
- How Schedulers Define Output Queue Properties on page 254

Define Scheduler Maps

Use scheduler maps to map schedulers to forwarding classes. For example:

```
[edit class-of-service]
scheduler-maps {
  sched1 {
    forwarding-class data-queue scheduler data-scheduler;
    forwarding-class video-queue scheduler video-scheduler;
    forwarding-class voice-queue scheduler voice-scheduler;
    forwarding-class nc-queue scheduler nc-scheduler;
  }
}
```

See Also
- Configuring Scheduler Maps on page 258

Define CoS Header Rewrite Rules

Use rewrite rules to redefine the CoS bit pattern of outgoing packets. For example:

```
[edit class-of-service]
rewrite-rules {
  inet-precedence inet-rewrite {
    forwarding-class data-queue {
      loss-priority low code-point 000;
      loss-priority high code-point 001;
    }
    forwarding-class voice-queue {
      loss-priority low code-point 010;
      loss-priority high code-point 011;
    }
    forwarding-class video-queue {
      loss-priority low code-point 100;
      loss-priority high code-point 101;
    }
    forwarding-class nc-queue {
      loss-priority low code-point 110;
      loss-priority high code-point 111;
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
  }
}
```

See Also

- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388

Apply Scheduler Maps and Rewrite Rules to Egress Interfaces

```
[edit class-of-service]
interfaces {
  ge-* {
    scheduler-map sched1;
    unit * {
      rewrite-rules {
        inet-precedence inet-rewrite;
      }
    }
  }
}
```

See Also

- Applying Scheduler Maps Overview on page 259
- Applying Rewrite Rules to Output Logical Interfaces on page 401

Related Documentation

- Packet Flow Through the Junos OS CoS Process Overview on page 17

Example: Classifying All Traffic from a Remote Device by Configuring Fixed Interface-Based Classification

This example shows the configuration of fixed classification based on the incoming interface. Fixed classification can be based on the physical interface (such as an ATM or Gigabit Ethernet interface) or a logical interface (such as an Ethernet VLAN, a Frame Relay DLCI, or an MPLS tunnel).

- Requirements on page 25
- Overview on page 26
- Configuration on page 27
- Verification on page 30

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on SRX Series devices running Junos OS Release 12.1. The SRX devices are configured to run as routers.

Copyright © 2019, Juniper Networks, Inc.

TIP: If you are performing tests on SRX devices, you might need to configure the devices to run as unsecured routers in your test environment. You would not typically do this in a production environment.

Overview

A fixed interface classifier is the simplest way to classify all packets from a specific interface to a forwarding class. This is typically used on edge routers to classify all traffic from a remote router or server to a certain forwarding class and queue. A fixed interface classifier simply looks at the ingress interface on which the packet arrives and assigns all traffic received on that interface to a certain class of service.

The fixed interface classifier cannot set the locally-meaningful packet-loss-priority, which is used by rewrite rules and drop profiles. The implicit packet-loss-priority is low for all fixed interface classifiers.

A fixed interface classifier is inadequate for scenarios in which interfaces receive traffic that belongs to multiple classes of service. However, interface-based classification can be useful when it is combined with other classification processes. Filtering based on the inbound interface can improve the granularity of classification, for example, when combined with filtering based on code point markings. Combining the processes for interface and code point marking classification allows a single code point marking to have different meanings, depending on the interface on which the packet is received. If you want to combine a fixed interface classifier with a code point classifier, this is in effect a multifield classifier.

More Granular Alternative to Fixed Interface Classifier

In Junos OS, you can combine interface-based classification and code-point classification by using a multifield classifier, as follows:

```
[edit firewall family inet filter MF_CLASSIFIER term 1]
from {
   dscp ef;
   interface ge-0/0/0.0;
}
then forwarding-class Voice;
```

Classifiers are described in more detail in the following Juniper Networks Learning Byte video.

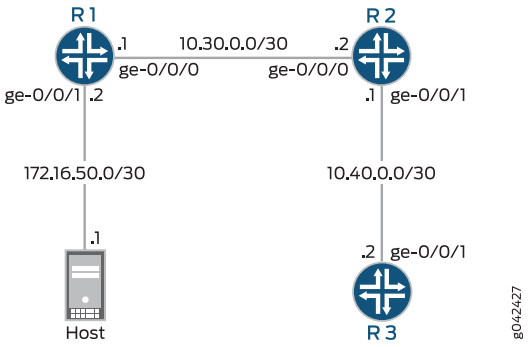
Video: Class of Service Basics, Part 2: Classification Learning Byte

Topology

Figure 8 on page 27 shows the sample network.

Copyright © 2019, Juniper Networks, Inc.

Figure 8: Fixed-Interface Classifier Scenario



To simulate voice traffic, this example shows TCP packets sent from the host to a downstream device. On Device R2, a fixed interface classifier routes the packets into the queue defined for voice traffic.

The classifier is assigned to interface ge-0/0/0 on Device R2. As always, verification of queue assignment is done on the egress interface, which is ge-0/0/1 on Device R2.

"CLI Quick Configuration" on page 27 shows the configuration for all of the Juniper Networks devices in Figure 8 on page 27. The section "Step-by-Step Procedure" on page 28 describes the steps on Device R2.

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.

Device R1

```
set interfaces ge-0/0/0 description to-R2
set interfaces ge-0/0/0 unit 0 family inet address 10.30.0.1/30
set interfaces ge-0/0/1 description to-host
set interfaces ge-0/0/1 unit 0 family inet address 172.16.50.2/30
set interfaces lo0 unit 0 family inet address 192.168.0.1/32
set protocols ospf area 0.0.0.0 interface ge-0/0/0.0
set protocols ospf area 0.0.0.0 interface ge-0/0/1.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
```

Device R2

```
set interfaces ge-0/0/0 unit 0 family inet address 10.30.0.2/30
set interfaces ge-0/0/1 unit 0 family inet address 10.40.0.1/30
set interfaces lo0 unit 0 family inet address 192.168.0.2/32
set protocols ospf area 0.0.0.0 interface ge-0/0/0.0
set protocols ospf area 0.0.0.0 interface ge-0/0/1.0
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set class-of-service forwarding-classes queue 0 BE-data
set class-of-service forwarding-classes queue 1 Premium-data
set class-of-service forwarding-classes queue 2 Voice
```

Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service forwarding-classes queue 3 NC
set class-of-service interfaces ge-0/0/0 unit 0 forwarding-class Voice
```

Device R3

```
set interfaces ge-0/0/0 unit 0 family inet address 10.50.0.1/30
set interfaces ge-0/0/1 unit 0 family inet address 10.40.0.2/30
set interfaces lo0 unit 0 family inet address 192.168.0.3/32
set protocols ospf area 0.0.0.0 interface ge-0/0/0.0
set protocols ospf area 0.0.0.0 interface ge-0/0/1.0
set protocols ospf area 0.0.0.0 interface lo0.0 passive
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To enable the default DSCP behavior aggregate classifier:

1. Configure the device interfaces.

```
[edit interfaces]
user@R2# set ge-0/0/0 unit 0 family inet address 10.30.0.2/30
user@R2# set ge-0/0/1 unit 0 family inet address 10.40.0.1/30
user@R2# set lo0 unit 0 family inet address 192.168.0.2/32
```

2. Configure an interior gateway protocol (IGP) or static routes.

```
[edit protocols ospf area 0.0.0.0]
user@R2# set interface ge-0/0/0.0
user@R2# set interface ge-0/0/1.0
user@R2# set interface lo0.0 passive
```

3. Configure a set of forwarding classes.

```
[edit class-of-service forwarding-classes]
user@R2# set queue 0 BE-data
user@R2# set queue 1 Premium-data
user@R2# set queue 2 Voice
user@R2# set queue 3 NC
```

4. Map all traffic that arrives on ge-0/0/0.0 into the Voice queue.

```
[edit class-of-service interfaces ge-0/0/0 unit 0]
user@R2# set forwarding-class Voice
```

Results

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

Copyright © 2019, Juniper Networks, Inc.

```
user@R2# show interfaces
ge-0/0/0 {
   unit 0 {
      family inet {
         address 10.30.0.2/30;
      }
   }
}
ge-0/0/1 {
   unit 0 {
      family inet {
         address 10.40.0.1/30;
      }
   }
}
lo0 {
   unit 0 {
      family inet {
         address 192.168.0.2/32;
      }
   }
}
```

```
user@R2# show protocols
ospf {
   area 0.0.0.0 {
      interface ge-0/0/0.0;
      interface ge-0/0/1.0;
      interface lo0.0 {
         passive;
      }
   }
}
```

```
user@R2# show class-or-service
forwarding-classes {
   queue 0 BE-data;
   queue 1 Premium-data;
   queue 2 Voice;
   queue 3 NC;
}
interfaces {
   ge-0/0/0 {
      unit 0 {
         forwarding-class Voice;
      }
   }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Copyright © 2019, Juniper Networks, Inc.

Verification

Confirm that the configuration is working properly.

- Verifying a Fixed-Interface Classifier on page 30

Verifying a Fixed-Interface Classifier

Purpose Verify that the fixed interface classifier is enabled on the Device R2's ingress interface. Keep in mind that although the classifier operates on incoming packets, you view the resulting queue assignment on the outgoing (egress) interface.

Action

1. Clear the interface statistics on Device R2's egress interface.

   ```
   user@R2> clear interface statistics ge-0/0/1
   ```

2. Using a packet generator, send TCP packets to a device that is downstream of Device R2.

   This example uses the packet generator hping.

   ```
   root@host> sudo hping3 10.40.0.2 -c 25 –fast

   HPING 10.40.0.2 (eth0 10.40.0.2): NO FLAGS are set, 40 headers + 0 data bytes
   len=46 ip=10.40.0.2 ttl=62 id=8619 sport=0 flags=RA seq=0 win=0 rtt=1.9 ms
   len=46 ip=10.40.0.2 ttl=62 id=8620 sport=0 flags=RA seq=1 win=0 rtt=2.8 ms
   len=46 ip=10.40.0.2 ttl=62 id=8621 sport=0 flags=RA seq=2 win=0 rtt=1.9 ms
   len=46 ip=10.40.0.2 ttl=62 id=8623 sport=0 flags=RA seq=3 win=0 rtt=1.8 ms
   len=46 ip=10.40.0.2 ttl=62 id=8624 sport=0 flags=RA seq=4 win=0 rtt=7.1 ms
   len=46 ip=10.40.0.2 ttl=62 id=8625 sport=0 flags=RA seq=5 win=0 rtt=1.8 ms
   len=46 ip=10.40.0.2 ttl=62 id=8626 sport=0 flags=RA seq=6 win=0 rtt=1.8 ms
   len=46 ip=10.40.0.2 ttl=62 id=8627 sport=0 flags=RA seq=7 win=0 rtt=1.9 ms
   len=46 ip=10.40.0.2 ttl=62 id=8628 sport=0 flags=RA seq=8 win=0 rtt=2.0 ms
   len=46 ip=10.40.0.2 ttl=62 id=8634 sport=0 flags=RA seq=9 win=0 rtt=7.4 ms
   len=46 ip=10.40.0.2 ttl=62 id=8635 sport=0 flags=RA seq=10 win=0 rtt=1.8 ms
   len=46 ip=10.40.0.2 ttl=62 id=8636 sport=0 flags=RA seq=11 win=0 rtt=2.0 ms
   len=46 ip=10.40.0.2 ttl=62 id=8637 sport=0 flags=RA seq=12 win=0 rtt=7.8 ms
   len=46 ip=10.40.0.2 ttl=62 id=8639 sport=0 flags=RA seq=13 win=0 rtt=7.0 ms
   len=46 ip=10.40.0.2 ttl=62 id=8640 sport=0 flags=RA seq=14 win=0 rtt=1.8 ms
   len=46 ip=10.40.0.2 ttl=62 id=8641 sport=0 flags=RA seq=15 win=0 rtt=7.2 ms
   len=46 ip=10.40.0.2 ttl=62 id=8642 sport=0 flags=RA seq=16 win=0 rtt=2.1 ms
   len=46 ip=10.40.0.2 ttl=62 id=8643 sport=0 flags=RA seq=17 win=0 rtt=2.0 ms
   len=46 ip=10.40.0.2 ttl=62 id=8644 sport=0 flags=RA seq=18 win=0 rtt=7.3 ms
   len=46 ip=10.40.0.2 ttl=62 id=8645 sport=0 flags=RA seq=19 win=0 rtt=1.7 ms
   len=46 ip=10.40.0.2 ttl=62 id=8646 sport=0 flags=RA seq=20 win=0 rtt=7.1 ms
   len=46 ip=10.40.0.2 ttl=62 id=8647 sport=0 flags=RA seq=21 win=0 rtt=2.0 ms
   len=46 ip=10.40.0.2 ttl=62 id=8648 sport=0 flags=RA seq=22 win=0 rtt=1.7 ms
   len=46 ip=10.40.0.2 ttl=62 id=8649 sport=0 flags=RA seq=23 win=0 rtt=1.8 ms
   len=46 ip=10.40.0.2 ttl=62 id=8651 sport=0 flags=RA seq=24 win=0 rtt=1.8 ms
   ```

3. On Device R2, verify that the Voice queue is incrementing.

   ```
   user@R2> show interfaces extensive ge-0/0/1 | find "queue counters"
   ```


Copyright © 2019, Juniper Networks, Inc.

```
Queue counters:      Queued packets  Transmitted packets     Dropped packets

  0 BE-data                      0                    0
0
  1 Premium-data                 0                    0
0
  2 Voice                       25                   25
0
  3 NC                           3                    3
0
Queue number:         Mapped forwarding classes
  0                   BE-data
  1                   Premium-data
  2                   Voice
  3                   NC
...
```

Meaning The output shows that the Voice queue has incremented by 25 packets after sending 25 packets through the ge-0/0/0 interface on Device R2.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Default Aliases for CoS Value Bit Patterns Overview on page 45
- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351

Interface Types That Do Not Support Junos OS CoS

For original Channelized OC12 PICs, limited CoS functionality is supported. For more information, contact Juniper Networks customer support.

The standard Junos OS CoS hierarchy is not supported on ATM interfaces. ATM has traffic-shaping capabilities that would override CoS, because ATM traffic shaping is performed at the ATM layer and CoS is performed at the IP layer. For more information about ATM traffic shaping and ATM CoS components, see the *Junos OS Network Interfaces Library for Routing Devices*.

NOTE: **Transmission scheduling is not supported on 8-port, 12-port, and 48-port Fast Ethernet PICs.**

You can configure CoS on all interfaces, except the following:

- **cau4**—Channelized STM1 IQ interface (configured on the Channelized STM1 IQ PIC).
- **coc1**—Channelized OC1 IQ interface (configured on the Channelized OC12 IQ PIC).
- **coc12**—Channelized OC12 IQ interface (configured on the Channelized OC12 IQ PIC).
- **cstm-1**—Channelized STM1 IQ interface (configured on the Channelized STM1 IQ PIC).

Copyright © 2019, Juniper Networks, Inc.

- **ct1**—Channelized T1 IQ interface (configured on the Channelized DS3 IQ PIC or Channelized OC12 IQ PIC).
- **ct3**—Channelized T3 IQ interface (configured on the Channelized DS3 IQ PIC or Channelized OC12 IQ PIC).
- **ce1**—Channelized E1 IQ interface (configured on the Channelized E1 IQ PIC or Channelized STM1 IQ PIC).
- **dsc**—Discard interface.
- **fxp**—Management and internal Ethernet interfaces.
- **lo**—Loopback interface. This interface is internally generated.
- **pe**—Encapsulates packets destined for the rendezvous point router. This interface is present on the first-hop router.
- **pd**—De-encapsulates packets at the rendezvous point. This interface is present on the rendezvous point.
- **vt**—Virtual loopback tunnel interface.

NOTE: For channelized interfaces, you can configure CoS on channels, but not at the controller level.

Related Documentation

- CoS on ATM Interfaces Overview on page 841

Copyright © 2019, Juniper Networks, Inc.

PART 2

Configuring Class of Service

- Assigning Service Levels with Behavior Aggregate Classifiers on page 35
- Assigning Service Levels with Multifield Classifiers on page 97
- Controlling Network Access with Traffic Policing on page 115
- Defining Forwarding Behavior with Forwarding Classes on page 205
- Defining Output Queue Properties with Schedulers on page 253
- Controlling Bandwidth with Scheduler Rates on page 273
- Setting Transmission Order with Scheduler Priorities and Hierarchical Scheduling on page 327
- Controlling Congestion with Scheduler RED Drop Profiles and Buffers on page 351
- Altering Outgoing Packet Headers Using Rewrite Rules on page 387
- Altering Class of Service Values in Packets Exiting the Network Using IPv6 DiffServ on page 475

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 2

Assigning Service Levels with Behavior Aggregate Classifiers

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Default IP Precedence Classifier on page 39
- Default DSCP and DSCP IPv6 Classifiers on page 41
- Default MPLS EXP Classifier on page 42
- Default IEEE 802.1p Classifier on page 43
- Default IEEE 802.1ad Classifier on page 44
- Default Aliases for CoS Value Bit Patterns Overview on page 45
- Defining Aliases for CoS Value Bit Patterns on page 48
- Configuring Behavior Aggregate Classifiers on page 51
- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53
- Example: Configuring and Applying a Default DSCP Behavior Aggregate Classifier on page 58
- Example: Configuring Behavior Aggregate Classifiers on page 65
- Understanding DSCP Classification for VPLS on page 75
- Example: Configuring DSCP Classification for VPLS on page 76
- Configuring Class of Service for MPLS LSPs on page 79
- Applying DSCP Classifiers to MPLS Traffic on page 83
- Applying MPLS EXP Classifiers to Routing Instances on page 87
- Applying MPLS EXP Classifiers for Explicit-Null Labels on page 94

Copyright © 2019, Juniper Networks, Inc.

Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic

The idea behind class of service (CoS) is that packets are not treated identically by the routers or switches on the network. In order to selectively apply service classes to specific packets, the packets of interest must be classified in some fashion.

The simplest way to classify a packet is to use behavior aggregate (BA) classification, also called the CoS value in this document. The DSCP, DSCP IPv6, or IP precedence bits of the IP header convey the behavior aggregate class information. The information might also be found in the MPLS EXP bits, IEEE 802.1ad, or IEEE 802.1p CoS bits.

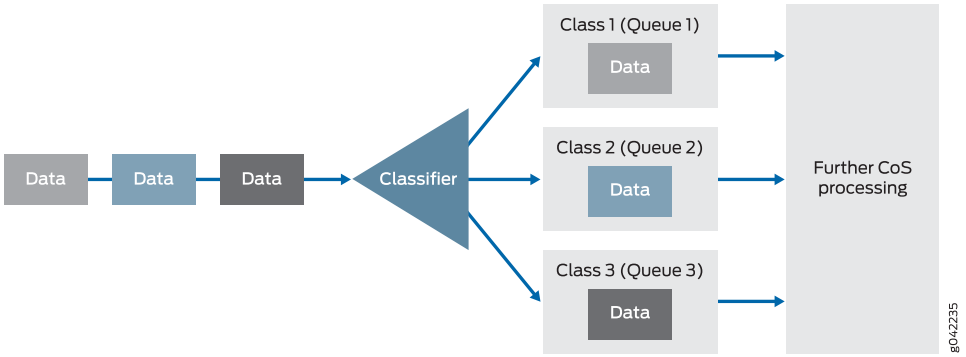
NOTE: Support was added for filtering on Differentiated Services Code Point (DSCP) and forwarding class for Routing Engine sourced packets, including IS-IS packets encapsulated in generic routing encapsulation (GRE). Subsequently, when upgrading from a previous version of Junos OS where you have both a class of service (CoS) and firewall filter, and both include DSCP or forwarding class filter actions, the criteria in the firewall filter automatically takes precedence over the CoS settings. The same is true when creating new configurations; that is, where the same settings exist, the firewall filter takes precedence over the CoS, regardless of which was created first.

BA classification is useful if the traffic comes from a trusted source and the CoS value in the packet header is trusted. If the traffic is untrusted, multifield classifiers (see "Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields" on page 97) are used to classify packets based on multiple packet fields. It is common to use multifield classifiers to classify traffic at the ingress of a network, rewrite the packet headers (see "Rewriting Packet Headers to Ensure Forwarding Behavior" on page 388), then use the more efficient BA classification for transversing the network.

The BA classifier maps a CoS value in the packet header to a forwarding class and loss priority. The forwarding class determines the output queue. The loss priority is used by schedulers in conjunction with the random early detection (RED) algorithm to control packet discard during periods of congestion.

Figure 9 on page 36 provides a high-level illustration of how a classifier works.

Figure 9: How a Classifier Works



Copyright © 2019, Juniper Networks, Inc.

The types of BA classifiers are based on which part of the incoming packet the classifier examines:

- DSCP, DSCP IPv6, or IP precedence—IP packet classification (Layer 3 headers)
- MPLS EXP—MPLS packet classification (Layer 2 headers)
- IEEE 802.1p—Packet classification (Layer 2 headers)
- IEEE 802.1ad—Packet classification for IEEE 802.1ad formats (including DEI bit)

Unlike multifield classifiers (which are discussed in "Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields" on page 97), BA classifiers are based on fixed-length fields, which makes them computationally more efficient than multifield classifiers. For this reason, core devices are normally configured to perform BA classification, because of the higher traffic volumes they handle.

In most cases, you need to rewrite a given marker (IP precedence, DSCP, IEEE 802.1p, IEEE 802.1ad, or MPLS EXP settings) at the ingress node to accommodate BA classification by core and egress devices. For more information about rewrite markers, see "Rewriting Packet Headers to Ensure Forwarding Behavior" on page 388.

NOTE: If you apply an IEEE 802.1 classifier to a logical interface, this classifier takes precedence and is not compatible with any other classifier type. Classifiers for IP (DSCP or IP precedence) and MPLS (EXP) can coexist on a logical interface if the hardware requirements are met.

For Juniper Networks M Series Multiservice Edge Routers, four classes can forward traffic independently. For M320 Multiservice Edge Routers, T Series Core Routers, MX Series 5G Universal Routing Platforms, and PTX Series Packet Transport Routers, eight classes can forward traffic independently. If you carry more classes of traffic than the device can forward independently, you must configure the additional classes to be aggregated into one of the available classes. You use the BA classifier to configure class aggregation.

Copyright © 2019, Juniper Networks, Inc.

NOTE: For a specified interface, you can configure both a multifield classifier and a BA classifier without conflicts. Because the classifiers are applied in sequential order if they are both either protocol specific or protocol independent, the BA classifier followed by the multifield classifier, any BA classification result is overridden by a multifield classifier if they conflict.

In the case that a protocol-specific BA classifier and a protocol-independent firewall filter are both configured together, the protocol-independent filter is processed before the protocol-specific BA classifier, regardless or protocol family. **firewall family any filter** is protocol independent and will be always processed before protocol-specific BA classifiers.

Fixed classification is protocol independent as well, hence, it is executed before any firewall filter.

For more information about multifield classifiers, see "Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields" on page 97. For more information about protocol-independent filters, see *Guidelines for Configuring Firewall Filters*. For more information about fixed classification, see "Applying Forwarding Classes to Interfaces" on page 231.

If you do nothing to configure or assign classifiers, Junos OS automatically assigns an implicit default IP precedence classifier to all logical interfaces that maps IP precedence code points to **best-effort** and **network-control** forwarding classes (mapped to queue 0 and queue 3 on routing devices, respectively). The default Junos OS CoS policy reserves 5 percent of available bandwidth for **network-control** traffic and 95 percent for **best-effort** traffic. Junos OS provides a range of default BA classifiers that you can apply to logical interfaces and that map various CoS values to **assured-forwarding** and **expedited-forwarding** forwarding classes as well as to the **best-effort** and **network-control** forwarding classes. You can also define custom BA classifiers that map any CoS value to any classifier you define.

NOTE: The default Junos OS CoS policy, 95 percent of the bandwidth for queue 0 and 5 percent for queue 3 on routing devices (see "Default Schedulers Overview" on page 257), is in effect regardless of any custom BA classifier or forwarding class definitions, until you configure a custom scheduler (see "Configuring Schedulers" on page 258).

If you enable the MPLS protocol family on a logical interface, a default MPLS EXP classifier is automatically applied to that logical interface. This default EXP classifier (see "Default MPLS EXP Classifier" on page 42) maps the eight possible EXP code point values into a combination of the four default forwarding classes and loss priority values to be directly compatible with the default EXP rewrite rule (see "Rewriting MPLS and IPv4 Packet Headers" on page 403).

Other default classifiers (such as those for IEEE 802.1p bits and DSCP) require that you explicitly associate a default classification table with a logical interface. When you

 Copyright © 2019, Juniper Networks, Inc.

explicitly associate a default classifier with a logical interface, you are in effect overriding the implicit default classifier with an explicit default classifier.

NOTE: Only the IEEE 802.1p classifier is supported in Layer 2-only interfaces. You must explicitly apply this classifier to the interface as shown in "Default IEEE 802.1p Classifier" on page 43.

NOTE: Although several CoS values map to the expedited-forwarding (ef) and assured-forwarding (af) classes, by default no resources are assigned to these forwarding classes. All af classes other than af1x are mapped to best-effort, because RFC 2597, *Assured Forwarding PHB Group*, prohibits a node from aggregating classes.

You can apply IEEE 802.1p classifiers to interfaces that are part of VPLS routing instances.

Release History Table

Release	Description
13.3R7	Support was added for filtering on Differentiated Services Code Point (DSCP) and forwarding class for Routing Engine sourced packets, including IS-IS packets encapsulated in generic routing encapsulation (GRE).

Related Documentation

- Default IP Precedence Classifier on page 39
- Default DSCP and DSCP IPv6 Classifiers on page 41
- Default MPLS EXP Classifier on page 42
- Default IEEE 802.1p Classifier on page 43
- Default IEEE 802.1ad Classifier on page 44
- Configuring Behavior Aggregate Classifiers on page 51
- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53
- Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 97
- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388

Default IP Precedence Classifier

By default, all logical interfaces are automatically assigned an implicit IP precedence classifier called **ipprec-compatibility**. The **ipprec-compatibility** IP precedence classifier maps IP precedence bits to forwarding classes and packet loss priorities (PLPs), as shown in Table 4 on page 40.

Copyright © 2019, Juniper Networks, Inc.

Table 4: Default IP Precedence (ipprec-compatibility) Classifier

IP Precedence Bits	Forwarding Class	Loss Priority
000	best-effort	low
001	best-effort	high
010	best-effort	low
011	best-effort	high
100	best-effort	low
101	best-effort	high
110	network-control	low
111	network-control	high

The other default IP precedence classifier (called **ipprec-default**) overrides the **ipprec-compatibility** classifier when you explicitly associate it with a logical interface. To do this, include the **default** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* classifiers inet-precedence]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number classifiers
   inet-precedence]
default;
```

Table 5 on page 40 shows the forwarding class and PLP that are assigned to the IP precedence bits when you apply the default IP precedence classifier.

Table 5: Default IP Precedence (ipprec-default) Classifier

IP Precedence Bits	Forwarding Class	PLP
000	best-effort	low
001	assured-forwarding	low
010	best-effort	low
011	best-effort	low
100	best-effort	low
101	expedited-forwarding	low
110	network-control	low
111	network-control	high

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53

Default DSCP and DSCP IPv6 Classifiers

To enable the default DiffServ code point (DSCP) classifier, include the **default** statement at the **[edit class-of-service interfaces *interface-name* unit *unit-number* classifiers dscp]** hierarchy level.

To enable the default DSCP IPv6 classifier, include the **default** statement at the **[edit class-of-service interfaces *interface-name* unit *unit-number* classifiers dscp-ipv6]** hierarchy level.

NOTE: If you deactivate or delete the **dscp-ipv6** statement from the configuration, the default IPv6 classifier is not activated on the M5, M10, M7i, M10i, M20, M40, M40e, and M160 routing platforms. As a workaround, explicitly specify the default option to the **dscp-ipv6** statement.

Table 6 on page 41 shows the forwarding class and packet loss priority (PLP) that are assigned to each well-known DSCP when you apply the explicit default DSCP or DSCP IPv6 classifier.

Table 6: Default DSCP and DSCP IPv6 Classifiers

DSCP and DSCP IPv6 Code Point	Forwarding Class	PLP
000000	best-effort	low
001010	assured-forwarding	low
001100	assured-forwarding	high
001110	assured-forwarding	high
101110	expedited-forwarding	low
110000	network-control	low
111000	network-control	low
all other code points	best-effort	low

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Default Aliases for CoS Value Bit Patterns Overview on page 45
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 241

Copyright © 2019, Juniper Networks, Inc.

- classifiers (Logical Interface) on page 1067

Default MPLS EXP Classifier

Multiprotocol Label Switching (MPLS) class of service (CoS) works in conjunction with the routing device's general CoS functionality.

When IP traffic enters a label-switched path (LSP) tunnel, the ingress routing device marks all packets with a class-of-service (CoS) value, which is used to place the traffic into a transmission queue. On the routing device, each physical interface has up to eight transmission queues. The CoS value is encoded as part of the MPLS header and remains in the packets until the MPLS header is removed when the packets exit from the egress routing device. The routing devices within the LSP utilize the CoS value set at the ingress routing device. The CoS value is encoded by means of the CoS bits (also known as the EXP or experimental bits).

If you do not configure any CoS features, the default general CoS settings are used. For MPLS class of service, you might want to prioritize how the transmission queues are serviced by configuring weighted round-robin, and to configure congestion avoidance using random early detection (RED).

For all PICs except PICs mounted on Juniper Networks M Series Multiservice Edge Router standard (nonenhanced) FPCs, if you enable the MPLS protocol family on a logical interface, the default MPLS EXP classifier is automatically applied to that logical interface.

Table 7 on page 42 lists the default MPLS classifier mapping of EXP bits to forwarding classes and loss priorities..

Table 7: Default MPLS EXP Classification

MPLS EXP Bits	Forwarding Class	Loss Priority
000	best-effort	low
001	best-effort	high
010	expedited-forwarding	low
011	expedited-forwarding	high
100	assured-forwarding	low
101	assured-forwarding	high
110	network-control	low
111	network-control	high

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring Class of Service for MPLS LSPs on page 79
- Default Aliases for CoS Value Bit Patterns Overview on page 45
- code-point-aliases on page 1072

Default IEEE 802.1p Classifier

Table 8 on page 43 shows the forwarding class and PLP that are assigned to each IEEE 802.1p CoS value when you apply the explicit default IEEE 802.1p classifier. To do this, include the **default** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* classifiers ieee-802.1]** hierarchy level:

NOTE: Only the IEEE 802.1p classifier is supported in Layer 2 interfaces. You must explicitly apply this classifier as shown.

```
[edit class-of-service interfaces interface-name unit logical-unit-number classifiers
   ieee-802.1]
default;
```

Table 8: Default IEEE 802.1p Classifier

IEEE 802.1p CoS Value	Forwarding Class	PLP
000	best-effort	low
001	best-effort	high
010	expedited-forwarding	low
011	expedited-forwarding	high
100	assured-forwarding	low
101	assured-forwarding	high
110	network-control	low
111	network-control	high

Related Documentation

- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53
- Default IEEE 802.1ad Classifier on page 44

Copyright © 2019, Juniper Networks, Inc.

Default IEEE 802.1ad Classifier

Table 9 on page 44 shows the forwarding class and packet loss priority (PLP) that are assigned to each IEEE 802.1ad CoS value when you apply the explicit default IEEE 802.1ad classifier. The table is very similar to the IEEE 802.1p default table, but the loss priority is determined by the DEI bit. To apply the default table, include the **default** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* classifiers ieee-802.1]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number classifiers
   ieee-802.1ad]
default;
```

Table 9: Default IEEE 802.1ad Classifier

IEEE 802.1ad CoS Value	Forwarding Class	PLP
0000	best- effort	low
0001	best-effort	high
0010	best- effort	low
0011	best-effort	high
0100	expedited-forwarding	low
0101	expedited-forwarding	high
0110	expedited-forwarding	low
0111	expedited-forwarding	high
1000	assured-forwarding	low
1001	assured-forwarding	high
1010	assured-forwarding	low
1011	assured-forwarding	high
1100	network-control	low
1101	network-control	high
1110	network-control	low
1111	network-control	high

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring and Applying IEEE 802.1ad Classifiers on page 575

Default Aliases for CoS Value Bit Patterns Overview

Behavior aggregate (BA) classifiers use class-of-service (CoS) values—such as Differentiated Services code points (DSCPs), DSCP IPv6, IP precedence, IEEE 802.1, and MPLS experimental (EXP) bits—to associate incoming packets with a particular CoS servicing level (forwarding class and packet loss priority (PLP)). You can assign a meaningful name or alias to the CoS values and use this alias instead of bits when configuring CoS components. These aliases are not part of the specifications but are well known through usage. For example, the alias for DSCP 101110 is widely accepted as **ef** (expedited forwarding).

NOTE: CoS value aliases must begin with a letter and can be up to 64 characters long.

When you define classifiers, you can refer to the markers by alias names. You can configure user-defined classifiers in terms of alias names. If the value of an alias changes, it alters the behavior of any classifier that references it.

Table 10 on page 45 shows the default mappings between the CoS values and standard aliases.

Table 10: Default CoS Value Aliases

Default CoS Value Alias	CoS Value
DSCP and DSCP IPv6 CoS Aliases and CoS Values	
ef	101110
af11	001010
af12	001100
af13	001110
af21	010010
af22	010100
af23	010110
af31	011010
af32	011100
af33	011110

Copyright © 2019, Juniper Networks, Inc.

Table 10: Default CoS Value Aliases (continued)

Default CoS Value Alias	CoS Value
af41	100010
af42	100100
af43	100110
be	000000
cs1	001000
cs2	010000
cs3	011000
cs4	100000
cs5	101000
nc1/cs6	110000
nc2/cs7	111000
MPLS EXP CoS Aliases and CoS Values	
be	000
be1	001
ef	010
ef1	011
af11	100
af12	101
nc1/cs6	110
nc2/cs7	111
IEEE 802.1 CoS Aliases and CoS Values	
be	000
be1	001
ef	010

Copyright © 2019, Juniper Networks, Inc.

Table 10: Default CoS Value Aliases (continued)

Default CoS Value Alias	CoS Value
ef1	011
af11	100
af12	101
nc1/cs6	110
nc2/cs7	111
IEEE 802.1ad CoS Aliases and CoS Values	
be	0000
be-dei	0001
be1	0010
be1-dei	0011
ef	0100
ef-dei	0101
ef1	0110
ef1-dei	0111
af11	1000
af11-dei	1001
af12	1010
af12-dei	1011
nc1	1100
nc1-dei	1101
nc2	1110
nc2-dei	1111
Legacy IP Precedence CoS Aliases and CoS Values	
be	000

Copyright © 2019, Juniper Networks, Inc.

Table 10: Default CoS Value Aliases (continued)

Default CoS Value Alias	CoS Value
be1	001
ef	010
ef1	011
af11	100
af12	101
nc1/cs6	110
nc2/cs7	111

Related Documentation

- Defining Aliases for CoS Value Bit Patterns on page 48
- Default IP Precedence Classifier on page 39
- Default DSCP and DSCP IPv6 Classifiers on page 41
- Default MPLS EXP Classifier on page 43
- Default IEEE 802.1p Classifier on page 44
- Default IEEE 802.1ad Classifier on page 44
- **code-point-aliases on page 1072**

Defining Aliases for CoS Value Bit Patterns

To define a CoS value alias, include the **code-point-aliases** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
code-point-aliases {
   (dscp | dscp-ipv6 | exp | ieee-802.1 | ieee-802.1ad | inet-precedence) {
      alias-name bit-pattern;
   }
}
```

The CoS marker types are as follows:

- **dscp**—Differentiated Services code point aliases for IPv4 packets.
- **dscp-ipv6**—Differentiated Services code point aliases for IPv6 packets.
- **exp**—Layer 2 CoS values for MPLS packets.
- **ieee-802.1**—Layer 2 IEEE 802.1 CoS values.

Copyright © 2019, Juniper Networks, Inc.

- **ieee-802.1**—Layer 2 IEEE 802.1ad (DEI) CoS values.
- **inet-precedence**—IP precedence for IPv4 packets. IP precedence mapping requires only the first three bits of the DSCP field.

For example, you might configure the following aliases:

```
[edit class-of-service]
code-point-aliases {
   dscp {
      my1 110001;
      my2 101110;
      be 000001;
      cs7 110000;
   }
}
```

To specify this configuration:

1. Specify the code-point-alias type as DSCP:

   ```
   [edit]
   user@host# edit class-of-service code-point-aliases dscp
   ```

2. Specify the alias names and DSCP 6-bit pattern.

   ```
   [edit class-of-service code-point-aliases dscp]
   user@host# set my1 110001
   user@host# set my2 101110
   user@host# set be 000001
   user@host# set cs7 110000
   ```

This configuration produces the following mapping:

```
user@host> show class-of-service code-point-aliases dscp
Code point type: dscp

Alias             Bit pattern

ef/my2            101110

af11              001010

af12              001100

af13              001110

af21              010010

af22              010100
```

Copyright © 2019, Juniper Networks, Inc.

af23	010110
af31	011010
af32	011100
af33	011110
af41	100010
af42	100100
af43	100110
be	000001
cs1	001000
cs2	010000
cs3	011000
cs4	100000
cs5	101000
nc1/cs6/cs7	110000
nc2	111000
my1	110001

The following notes explain certain results in the mapping:

- **my1 110001**:
  - 110001 was not mapped to anything before, and **my1** is a new alias.
  - Nothing in the default mapping table is changed by this statement.
- **my2 101110**:
  - 101110 is now mapped to **my2** as well as **ef**.
- **be 000001**:
  - **be** is now mapped to 000001.

Copyright © 2019, Juniper Networks, Inc.

  - The old value of **be**, 000000, is not associated with any alias. Packets with this DSCP value are now mapped to the default forwarding class.

- **cs7 110000**:
  - **cs7** is now mapped to 110000, as well as **nc1** and **cs6**.
  - The old value of **cs7**, 111000, is still mapped to **nc2**.

Related Documentation

- Default Aliases for CoS Value Bit Patterns Overview on page 45
- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53

Configuring Behavior Aggregate Classifiers

You can override the default IP precedence classifier (**ipprec-compatibility**) by defining a custom behavior aggregate (BA) classifier and applying it to a logical interface or by applying one of the other default BA classifiers to a logical interface.

The BA classifiers map sets the forwarding class and packet loss priority (PLP) for a specific set of code-point aliases or bit patterns. The inputs of the map are CoS values aliases or bit patterns. The outputs of the map are the forwarding class and the PLP. For more information about how CoS maps work, see "Mapping CoS Component Inputs to Outputs" on page 11.

The classifiers work as follows:

- **dscp**—Handles incoming IPv4 packets.
- **dscp-ipv6**—Handles incoming IPv6 packets.
- **exp**—Handles MPLS packets using Layer 2 headers.
- **ieee-802.1**—Handles Layer 2 CoS.
- **ieee-802.1ad**—Handles IEEE 802.1ad formats (including DEI bit).
- **inet-precedence**—Handles incoming IPv4 packets. IP precedence mapping requires only the upper three bits of the DSCP field.

A classifier takes a specified Cos value as either the literal bit pattern or as a defined alias and attempts to match it to the type of packet arriving on the interface. If the information in the packet's header matches the specified pattern, the packet is sent to the appropriate queue, defined by the forwarding class associated with the classifier.

NOTE: On M Series, MX Series, and T Series routers, and EX Series switches that do not have tricolor marking enabled, the loss priority can be configured only by setting the PLP within a multifield classifier. This setting can then be used by the appropriate drop profile map and rewrite rule. For more information, see "Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows" on page 361.

Copyright © 2019, Juniper Networks, Inc.

Use the following configuration statements to define new classifiers for all CoS value types:

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

To define a new classifier for all CoS value types:

1. Specify the type and name of the new classifier. For example, to create a new DSCP type classifier called class1:

   ```
   [edit]
   user@host# edit class-of-service classifiers dscp class1
   ```

2. (Optional) Specify the forwarding class associated with the classifier.

   ```
   [edit class-of-service classifiers dscp class1]
   user@host# edit forwarding-class class-name
   ```

3. (Optional) Specify the packet loss priority (PLP) value and for a specific set of code-point aliases and bit patterns.

   ```
   [edit class-of-service classifiers dscp class1 forwarding-class best-effort]
   user@host# set loss-priority level code-points [ aliases ] [ bit-patterns]
   ```

When tricolor marking is enabled, four classifier PLP designations are supported: **low**, **medium-low**, **medium-high**, and **high**. For example, in the following configuration, the **assured-forwarding** forwarding class and **medium-low** PLP are assigned to all packets entering the interface with the **101110** CoS value:

1. Map the **assured-forwarding** forwarding class and **medium-low** PLP to the CoS value of **101110**.

   ```
   [edit class-of-service classifiers dscp class1]
   user@host# set forwarding-class assured forwarding loss-priority medium-low
     code-points 101110
   ```

2. Verify the configuration.

   ```
   [edit class-of-service classifiers dscp class1]
   user@host# show
   ```


Copyright © 2019, Juniper Networks, Inc.

```
forwarding-class assured-forwarding {
    loss-priority medium-low code-points 101110;
}
```

To use this classifier, you must configure the settings for the **assured-forwarding** forwarding class at the **[edit class-of-service forwarding-classes queue *queue-number* assured-forwarding]** hierarchy level. For more information, see "Understanding How Forwarding Classes Assign Classes to Output Queues" on page 205.

You can use any table, including the default, in the definition of a new classifier by including the **import** statement. The imported classifier is used as a template and is not modified. Whenever you commit a configuration that assigns a new ***class-name*** and **loss-priority** value to a CoS value alias or bit pattern, it replaces that entry in the imported classifier template. As a result, you must explicitly specify every CoS value in every designation that requires modification. For instance, to import the default DSCP classifier:

1. Specify the type and name of the new classifier. For example, to create a new DSCP type classifier called class1:

```
[edit]
user@host# edit class-of-service classifiers dscp class1
```

2. Specify the default DSCP classifier.

```
[edit class-of-service classifiers dscp class1]
user@host# set import default
```

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53
- Enabling Tricolor Marking and Limitations of Three-Color Policers on page 173

Applying Behavior Aggregate Classifiers to Logical Interfaces

This topic describes how to apply behavior aggregate (BA) classifiers to logical interfaces.

When you apply BA classifiers to a logical interface, you can use interface wildcards for the ***interface-name*** and ***logical-unit-number***.

For most PICs, if you apply an IEEE 802.1 classifier to a logical interface, you cannot apply non-IEEE classifiers to other logical interfaces on the same physical interface. This restriction does not apply to Gigabit Ethernet IQ2 PICs.

There are some restrictions on applying multiple BA classifiers to a single logical interface. Table 11 on page 54 shows the supported combinations. In this table, the OSE PICs refer to the 10-port 10-Gigabit OSE PICs.

Copyright © 2019, Juniper Networks, Inc.

Table 11: Logical Interface Classifier Combinations

Classifier Combinations	Gigabit Ethernet IQ2 PICs	OSE PICs	Other PICs on M320, MX Series, T Series routers and on EX Series Switches	Other M Series with Regular FPCs	Other M Series with Enhanced FPCs
dscp and **inet-precedence**	No	No	No	No	No
dscp-ipv6 and **(dscp | inet-precedence)**	Yes	Yes	Yes	No	No
exp and **ieee 802.1**	Yes	Yes	No	No	No
ieee 802.1 and **(dscp | dscp-ipv6 | exp | inet-precedence)**	Yes	Yes	No	No	Yes
exp and **(dscp | dscp-ipv6 | inet-precedence)**	Yes	Yes	Yes	No	Yes

For Gigabit Ethernet IQ2 and 10-port 10-Gigabit Oversubscribed Ethernet (OSE) interfaces, family-specific classifiers take precedence over IEEE 802.1p BA classifiers. For example, if you configure a logical interface to use both an MPLS EXP and an IEEE 802.1p classifier, the EXP classifier takes precedence. MPLS-labeled packets are evaluated by the EXP classifier, and all other packets are evaluated by the IEEE 802.1p classifier. The same is true about other classifiers when combined with IEEE 802.1p classifiers on the same logical interface.

NOTE: For an interface on an M Series FPC, you can apply only the default exp classifier. For an enhanced FPC, you can create a new exp classifier and apply it to an interface.

On MX960, MX480, MX240, MX80, M120, and M320 routers and EX Series switches with Enhanced Type III FPCs only, you can configure user-defined DSCP-based BA classification for MPLS interfaces (this feature is not available for IQE PICs or on MX Series routers and EX Series switches when ingress queuing is used) or VPLS or Layer 3 VPN routing instances (LSI interfaces). The DSCP-based classification for MPLS packets for Layer 2 VPNs is not supported.

Copyright © 2019, Juniper Networks, Inc.

NOTE: If you do not apply a DSCP classifier, the default EXP classifier is applied to MPLS traffic. At times you might need to maintain the original classifier of the incoming packet, where you neither want to configure a custom classifier for the interface nor accept the default classifier, which would override the original classifier. In that case, on MX Series devices only, you can apply the **no-default** option for the interface. For example:

```
[edit class-of-service]
interfaces interface-name unit unit-number {
  classifiers {
    no-default;
  }
}
```

You can apply DSCP classification for MPLS traffic in the following usage scenarios:

- In a Layer 3 VPN using a label-switched interface (LSI) routing instance.
  - Supported on the M120, M320, MX960, MX480, MX240, and MX80 routers.
  - DSCP classifier applied under **[edit class-of-service routing-instances]** on the egress provider edge (PE) router.
- In VPLS using an LSI routing instance.
  - Supported on the M120, M320, MX960, MX480, MX240, and MX80 routers.
  - DSCP classifier applied under **[edit class-of-service routing-instances]** on the egress PE router.
- In a Layer 3 VPN using a virtual tunnel (VT) routing instance.
  - Supported on the M120, M320, MX960, MX480, MX240, and MX80 routers.
  - DSCP classifier applied under **[edit class-of-service interfaces]** on the core-facing interface on the egress PE router.
- In VPLS using the VT routing instance.
- MPLS forwarding.
  - Supported on the M120, M320, MX960, MX480, MX240, and MX80 routers (not supported on IQE and MX when ingress queuing is enabled).
  - DSCP classifier applied under **[edit class-of-service interfaces]** on the ingress core-facing interface on the provider (P) or egress PE router.

MPLS forwarding when the label stacking is greater than 2 is not supported.

You can apply BA classifiers to a routing instance or a logical interface, depending on where you want to classify the packets:

- To classify MPLS packets on the routing instance at the egress PE, include the **dscp** or **dscp-ipv6** statements at the [**edit class-of-service routing-instances**

Copyright © 2019, Juniper Networks, Inc.

routing-instance-name **classifiers**] hierarchy level. For details, see "Applying MPLS EXP Classifiers to Routing Instances" on page 87.

- To classify MPLS packets at the core-facing interface, apply the classifier at the [**edit class-of-service interface** ***interface-name*** **unit** ***unit-name*** **classifiers (dscp | dscp-ipv6)** ***classifier-name*** **family mpls**] hierarchy level. The following procedure describes this method.

In the following example, you define a DSCP classifier for IPv4 named **dscp-ipv4-classifier** and a corresponding IPv6 DSCP classifier for the **fc-af11-class** forwarding class. You then apply the IPv4 classifier to MPLS traffic and the IPv6 classifier to Internet traffic on interface ge-2/0/3.0 or apply the same classifier to both MPLS and IP traffic on interface ge-2/2/0. This example shows both of these methods.

1. Define the IPv4 classifier.

```
[edit]
user@host# edit class-of-service
user@host# set classifiers dscp dscp-ipv4-classifier forwarding-class fc-af11-class
  loss-priority low code-points 000100
```

2. Define the IPv6 classifier.

```
[edit class-of-service]
user@host# set classifiers dscp-ipv6 dscp-ipv6-classifier forwarding-class fc-af11-class
  loss-priority low code-points af11
```

3. (Optional) Apply the IPv4 classifier to MPLS traffic and the IPv6 classifier to Internet traffic on interface ge-2/0/3.0.

```
[edit class-of-service]
user@host# set interfaces ge-2/0/3 unit 0 classifiers dscp dscp-ipv4-classifier family
  mpls
user@host# set interfaces ge-2/0/3 unit 0 classifiers dscp-ipv6 dscp-ipv6-classifier
  family inet
```

4. Confirm the configuration.

```
[edit class-of-service]
user@host# show
```

```
classifiers {
    dscp dscp-ipv4-classifier {
        forwarding-class fc-af11-class {
            loss-priority low code-points 000100;
        }
    }
    dscp-ipv6 dscp-ipv6-classifer {
        forwarding-class fc-af11-class {
            loss-priority low code-points af11;
        }
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
}
interfaces {
    ge-2/0/3 {
        unit 0 {
            classifiers {
                dscp dscp-ipv4-classifier {
                    family mpls;
                }
                dscp-ipv6 dscp-ipv6-classifier {
                    family inet;
                }
            }
        }
    }
}
```

5. (Optional) Apply the same classifier, named **dscp-mpls-and-inet**, to both MPLS and IP traffic on interface ge-2/2/0.

```
[edit class-of-service]
user@host# set interfaces ge-2/2/0 unit 0 classifiers dscp dscp-mpls-and-inet family
  [mpls inet]
```

6. Confirm the configuration.

```
[edit class-of-services interface ge-2/2/0]
user@host# show
```

```
unit 0 {
            classifiers {
                dscp dscp-mpls-and-inet {
                    family [ mpls inet ];
                }
            }
        }
```

NOTE: This is not a complete configuration.

NOTE: You can apply DSCP and DSCP IPv6 classifiers to explicit null MPLS packets. The family mpls statement works the same on both explicit null and non-null MPLS labels.

Related Documentation

- Applying DSCP Classifiers to MPLS Traffic on page 83

Copyright © 2019, Juniper Networks, Inc.

Example: Configuring and Applying a Default DSCP Behavior Aggregate Classifier

A Junos OS classifier identifies and separates traffic flows and provides the means to prioritize traffic later in the class-of-service (CoS) process.

A behavior aggregate (BA) classifier performs this function by associating well-known CoS values with forwarding classes and loss priorities. To enable a default classifier, you simply apply it to your device interfaces. If a default classifier is not applied to an interface, it does not take effect.

Junos OS provides multiple default BA classifier types, which you can combine and supplement with custom BA classifiers as needed to achieve your overall traffic classification goals. This example shows how to apply the default (BA) DiffServ code point (DSCP) classifier and verify its functionality.

- Requirements on page 58
- Overview on page 58
- Configuration on page 62
- Verification on page 63

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine. If you do not have access to a traffic generator, you can use extended ping for verification. This approach is shown as well.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Overview

The basis of Junos OS CoS is traffic differentiation. Assigning traffic to different classes of service provides the necessary differentiation. From the point of view of a router, the class of service assigned to a packet defines how the router behaves toward the packet. The concept of traffic differentiation is present in every CoS tool, and as a result, classes of service are present across the entire CoS design. A classifier has one input, the incoming packet, and it has *N* possible outputs, where *N* is the number of possible classes of service into which the packet can be classified.

BA classification is used when the traffic coming into your device already has trusted CoS values in the packet header. For example, the default DSCP BA classifier specifies that packets coming in with code points 000000 are assigned to the best-effort forwarding class and given a loss priority of low.

A forwarding class and loss priority are assigned by default to each well-known DSCP. To view this, run the show class-of-service classifier command.

```
user@host> show class-of-service classifier type dscp
```


Copyright © 2019, Juniper Networks, Inc.

```
Classifier: dscp-default, Code point type: dscp, Index: 7
  Code point         Forwarding class                    Loss priority
  000000             best-effort                         low
  000001             best-effort                         low
  000010             best-effort                         low
  000011             best-effort                         low
  000100             best-effort                         low
  000101             best-effort                         low
  000110             best-effort                         low
  000111             best-effort                         low
  001000             best-effort                         low
  001001             best-effort                         low
  001010             assured-forwarding                  low
  001011             best-effort                         low
  001100             assured-forwarding                  high
  001101             best-effort                         low
  001110             assured-forwarding                  high
  001111             best-effort                         low
  010000             best-effort                         low
  010001             best-effort                         low
  010010             best-effort                         low
  010011             best-effort                         low
  010100             best-effort                         low
  010101             best-effort                         low
  010110             best-effort                         low
  010111             best-effort                         low
  011000             best-effort                         low
  011001             best-effort                         low
  011010             best-effort                         low
  011011             best-effort                         low
  011100             best-effort                         low
  011101             best-effort                         low
  011110             best-effort                         low
  011111             best-effort                         low
  100000             best-effort                         low
  100001             best-effort                         low
  100010             best-effort                         low
  100011             best-effort                         low
  100100             best-effort                         low
  100101             best-effort                         low
  100110             best-effort                         low
  100111             best-effort                         low
  101000             best-effort                         low
  101001             best-effort                         low
  101010             best-effort                         low
  101011             best-effort                         low
  101100             best-effort                         low
  101101             best-effort                         low
  101110             expedited-forwarding                low
  101111             best-effort                         low
  110000             network-control                     low
  110001             best-effort                         low
  110010             best-effort                         low
  110011             best-effort                         low
  110100             best-effort                         low
  110101             best-effort                         low
  110110             best-effort                         low
  110111             best-effort                         low
  111000             network-control                     low
```

Copyright © 2019, Juniper Networks, Inc.

```
  111001            best-effort                    low
  111010            best-effort                    low
  111011            best-effort                    low
  111100            best-effort                    low
  111101            best-effort                    low
  111110            best-effort                    low
  111111            best-effort                    low
```

The forwarding class determines the output queue. By default, all best-effort traffic uses queue 0.

To view the queues that are associated, by default, with each forwarding class, use the **show class-of-service forwarding-class** command. (For clarity, some of the output is excluded.)

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

Forwarding class                        ID      Queue
  best-effort                            0       0
  expedited-forwarding                   1       1
  assured-forwarding                     2       2
  network-control                        3       3
```

The loss priority is used by schedulers in conjunction with the random early detection (RED) algorithm to control packet discard during periods of congestion. When you are thinking about loss priorities, keep in mind that unless you configure them, they have no meaning. The default drop behavior is to wait until the queue is 100 percent full and then begin dropping packets indiscriminately. When the queue dips below 100 percent full, packets stop dropping.

The default drop behavior is shown in the **show class-of-service drop-profile** command.

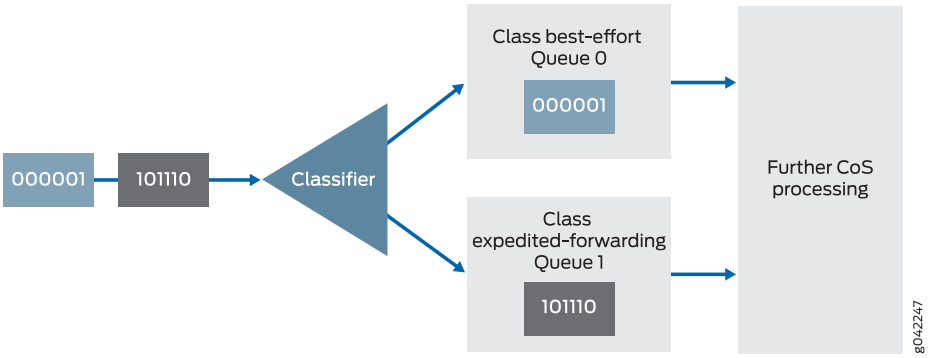
```
user@host> show class-of-service drop-profile

Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
```

To create meanings for the various loss priorities, you must configure custom drop profiles. For example, you might define the low loss priority to mean a 10 percent drop probability when the queue is 75 percent full and a 40 percent drop probability when the queue fill level is 95 percent. You might define the high loss priority to mean a 50 percent drop probability when the fill level is 25 percent and a 90 percent drop probability when the fill level is 50 percent. Custom drop profiles are not included in this example, but are mentioned here for clarity because classifiers assign loss priorities. It is important to understand that these assignments are meaningless until you create drop profiles.

The default classifier operation is shown in Figure 10 on page 61. The figure shows two IPv4 packets entering an interface and being classified according to the DSCP code points in the packet headers.

 Copyright © 2019, Juniper Networks, Inc.

Figure 10: Behavior Aggregate Classifier with Two Queues



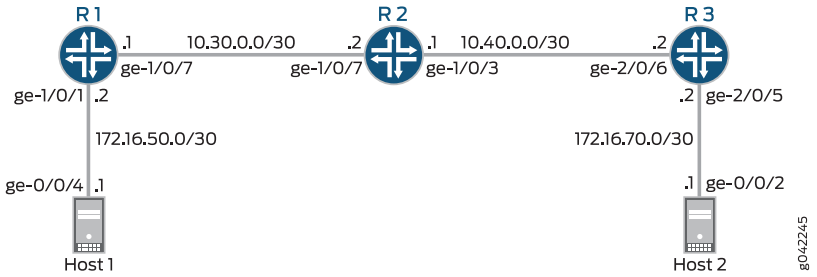
Classifiers are described in more detail in the following Juniper Networks Learning Byte video.

Video: Class of Service Basics, Part 2: Classification Learning Byte

Topology

Figure 11 on page 61 shows the sample network.

Figure 11: Behavior Aggregate Classifier Scenario



It is important to apply your class-of-service configuration across the topology, instead of applying it to a single device. Furthermore, even though classification takes effect on incoming interfaces, you should apply BA classifiers to all core and core-facing interfaces. This is because a single interface can be either incoming or outgoing, depending on the direction of the traffic. For example, as traffic flows from Host 1 to Host 2, the incoming interfaces are ge-1/0/7 on Device R2 and ge-2/0/6 on Device R3. As traffic flows in the other direction, from Host 2 to Host R1, the incoming interfaces are ge-1/0/3 on Device R2 and ge-1/0/7 on Device R1.

The BA classifier is not applied to ge-1/0/1 on Device R1 or ge-2/0/5 on Device R3, because these interfaces are not core facing. Generally, at the edge-facing interfaces, you would use a multifield classifier, not a BA classifier.

"CLI Quick Configuration" on page 62 shows the configuration for all of the Juniper Networks devices in Figure 11 on page 61. The section "Step-by-Step Procedure" on page 62 describes the steps on Device R2.

Copyright © 2019, Juniper Networks, Inc.

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.

Device R1

```
set interfaces ge-1/0/1 unit 0 family inet address 172.16.50.2/30
set interfaces ge-1/0/7 unit 0 family inet address 10.30.0.1/30
set class-of-service interfaces ge-1/0/9 unit 0 classifiers dscp default
```

Device R2

```
set interfaces ge-1/0/3 unit 0 family inet address 10.40.0.1/30
set interfaces ge-1/0/7 unit 0 family inet address 10.30.0.2/30
set class-of-service interfaces ge-1/0/3 unit 0 classifiers dscp default
set class-of-service interfaces ge-1/0/7 unit 0 classifiers dscp default
```

Device R3

```
set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
set interfaces ge-2/0/6 unit 0 family inet address 10.40.0.2/30
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To enable the default DSCP behavior aggregate classifier:

1. Configure the device interfaces.

```
[edit interfaces]
user@R2# set ge-1/0/3 unit 0 family inet address 10.40.0.1/30
user@R2# set ge-1/0/7 unit 0 family inet address 10.30.0.2/30
```

2. Enable the default DSCP classifier on the interfaces.

```
[edit class-of-service interfaces]
user@R2# set ge-1/0/3 unit 0 classifiers dscp default
user@R2# set ge-1/0/7 unit 0 classifiers dscp default
```

Results

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

```
user@R2# show interfaces
ge-1/0/3 {
   unit 0 {
      family inet {
```


Copyright © 2019, Juniper Networks, Inc.

```
        address 10.40.0.1/30;
      }
    }
  }
  ge-1/0/7 {
    unit 0 {
      family inet {
        address 10.30.0.2/30;
      }
    }
  }
```

```
user@R2# show class-or-service
interfaces {
  ge-1/0/3 {
    unit 0 {
      classifiers {
        dscp default;
      }
    }
  }
  ge-1/0/7 {
    unit 0 {
      classifiers {
        dscp default;
      }
    }
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

Verifying Behavior Aggregate Classifiers

Purpose Verify that the default behavior aggregate classifier is enabled on the device interfaces. Keep in mind that although the classifier operates on incoming packets, you view the resulting queue assignment on the outgoing interface.

Action

1. Clear the interface statistics on Device R2.

   ```
   user@R2> clear interface statistics ge-1/0/3
   ```

2. Using extended ping from Device R1 or a packet generator running on a host or server, send packets with the code point set to 001010.

   Both methods are shown here. The packet generator used is hping.

Copyright © 2019, Juniper Networks, Inc.

- When you are using extended ping to set the DSCP code points in the IPv4 packet header, the type-of-service (ToS) decimal value (in this case, 40) is required in the **tos** option of the **ping** command.
- When you are using hping to set the DSCP code points in the IPv4 packet header, the ToS hex value (in this case, 28) is required in the **--tos** option of the **hping** command.

If your binary-to-hex or binary-to-decimal conversion skills are rusty, you can use an online calculator, such as http://www.mathsisfun.com/binary-decimal-hexadecimal-converter.html .

NOTE: When you convert a binary DSCP code point value, be sure to add two extra zeros at the end. So instead of 001010, use 00101000. These 0 values (the 7th and 8th bits) are reserved and ignored, but if you do not include them in the conversion, your hex and decimal values will be incorrect.

```
user@R1> ping 172.16.70.1 tos 40 rapid count 25

PING 172.16.70.1 (172.16.70.1): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!
--- 172.16.70.1 ping statistics ---
25 packets transmitted, 25 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.430/0.477/0.847/0.079 ms
```

```
root@host1> hping 172.16.70.1 --tos 28 -c 25

HPING 172.16.70.1 (eth1 172.16.70.1): NO FLAGS are set, 40 headers + 0 data
bytes
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=0.3 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=1 win=0 rtt=0.6 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=2 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=3 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=4 win=0 rtt=0.6 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=5 win=0 rtt=0.3 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=6 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=7 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=8 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=9 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=10 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=11 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=12 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=13 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=14 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=15 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=16 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=17 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=18 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=19 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=20 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=21 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=22 win=0 rtt=0.4 ms
```


Copyright © 2019, Juniper Networks, Inc.

```
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=23 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=24 win=0 rtt=0.4 ms
```

3. On Device R2, verify that queue 2 is incrementing.

   Code point 001010 is associated with assured-forwarding, which uses queue 2 by default.

```
user@R2> show interfaces extensive ge-1/0/3 | find "queue counters"


Queue counters:     Queued packets  Transmitted packets      Dropped packets
    0                   0               0                       0
    1                   0               0                       0
    2                   50              25                      0
    3                   3               3                       0
  Queue number:         Mapped forwarding classes
    0                   best-effort
    1                   expedited-forwarding
    2                   assured-forwarding
    3                   network-control
```

Meaning The output shows that queue 2 has incremented by 50 packets after sending 50 packets through the router.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 361

Example: Configuring Behavior Aggregate Classifiers

This example shows how to configure behavior aggregate classifiers for a device to determine forwarding treatment of packets.

- Requirements on page 65
- Overview on page 66
- Configuration on page 67
- Verification on page 69

Requirements

Before you begin, determine the forwarding class and PLP that are assigned by default to each well-known DSCP that you want to configure for the behavior aggregate classifier. See *Default Behavior Aggregate Classification*.

Copyright © 2019, Juniper Networks, Inc.

Overview

You configure behavior aggregate classifiers to classify packets that contain valid DSCPs to appropriate queues. Once configured, you must apply the behavior aggregate classifier to the correct interfaces. You can override the default IP precedence classifier by defining a classifier and applying it to a logical interface. To define new classifiers for all code point types, include the **classifiers** statement at the **[edit class-of-service]** hierarchy level.

In this example, you set the DSCP behavior aggregate classifier to ba-classifier as the default DSCP map. You set a best-effort forwarding class as be-class, an expedited forwarding class as ef-class, an assured forwarding class as af-class, and a network control forwarding class as nc-class. Finally, you apply the behavior aggregate classifier to an interface called ge-0/0/0.

Table 12 on page 66 shows how the behavior aggregate classifier assigns loss priorities, to incoming packets in the four forwarding classes.

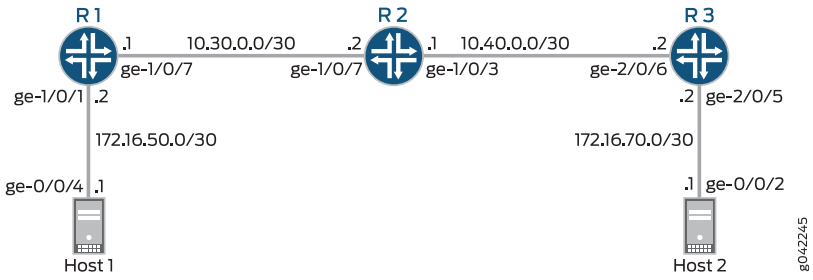
Table 12: Sample ba-classifier Loss Priority Assignments

mf-classifier Forwarding Class	For CoS Traffic Type	ba-classifier Assignments
be-class	Best-effort traffic	High-priority code point: **000001**
ef-class	Expedited forwarding traffic	High-priority code point: **101111**
af-class	Assured forwarding traffic	High-priority code point: **001100**
nc-class	Network control traffic	High-priority code point: **110001**

Topology

Figure 12 on page 66 shows the sample network.

Figure 12: Behavior Aggregate Classifier Scenario



"CLI Quick Configuration" on page 67 shows the configuration for all of the Juniper Networks devices in Figure 12 on page 66.

The section "Step-by-Step Procedure" on page 67 describes the steps on Device R2.

Copyright © 2019, Juniper Networks, Inc.

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from the configuration mode.

```
set class-of-service classifiers dscp ba-classifier import default
set class-of-service classifiers dscp ba-classifier forwarding-class be-class loss-priority
  high code-points 000001
set class-of-service classifiers dscp ba-classifier forwarding-class ef-class loss-priority
  high code-points 101111
set class-of-service classifiers dscp ba-classifier forwarding-class af-class loss-priority
  high code-points 001100
set class-of-service classifiers dscp ba-classifier forwarding-class nc-class loss-priority
  high code-points 110001
set class-of-service interfaces ge-0/0/0 unit 0 classifiers dscp ba-classifier
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the CLI User Guide.

To configure behavior aggregate classifiers for a device:

1. Configure the class of service.

   ```
   [edit]
   user@host# edit class-of-service
   ```

2. Configure behavior aggregate classifiers for DiffServ CoS.

   ```
   [edit class-of-service]
   user@host# edit classifiers dscp ba-classifier
   user@host# set import default
   ```

3. Configure a best-effort forwarding class classifier.

   ```
   [edit class-of-service classifiers dscp ba-classifier]
   user@host# set forwarding-class be-class loss-priority high code-points 000001
   ```

4. Configure an expedited forwarding class classifier.

   ```
   [edit class-of-service classifiers dscp ba-classifier]
   user@host# set forwarding-class ef-class loss-priority high code-points 101111
   ```

5. Configure an assured forwarding class classifier.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service classifiers dscp ba-classifier]
user@host# set forwarding-class af-class loss-priority high code-points 001100
```

6. Configure a network control forwarding class classifier.

```
[edit class-of-service classifiers dscp ba-classifier]
user@host# set forwarding-class nc-class loss-priority high code-points 110001
```

7. Apply the behavior aggregate classifier to an interface.

```
[edit]
user@host# set class-of-service interfaces ge-0/0/0 unit 0 classifiers dscp
   ba-classifier
```

NOTE: You can use interface wildcards for **interface-name** and **logical-unit-number**.

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

```
[edit]
user@host# show class-of-service
classifiers {
   dscp ba-classifier {
      import default;
      forwarding-class be-class {
         loss-priority high code-points 000001;
      }
      forwarding-class ef-class {
         loss-priority high code-points 101111;
      }
      forwarding-class af-class {
         loss-priority high code-points 001100;
      }
      forwarding-class nc-class {
         loss-priority high code-points 110001;
      }
   }
   forwarding-classes {
      class BE-data queue-num 0;
      class Premium-data queue-num 1;
      class Voice queue-num 2;
      class NC queue-num 3;
   }
   interfaces {
      ge-0/0/0 {
```


Copyright © 2019, Juniper Networks, Inc.

```
    unit 0 {
      classifiers {
        dscp ba-classifier;
      }
    }
  }
  ge-1/0/9 {
    unit 0 {
      classifiers {
        dscp v4-ba-classifier;
      }
      ge-1/0/9 {
        unit 0 {
          classifiers {
            dscp v4-ba-classifier;
          }
          ge-1/0/9 {
            unit 0 {
              classifiers {
                dscp v4-ba-classifier;
              }
            }
          }
        }
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Verifying the Code-Point Aliases on page 69
- Verifying the DSCP Classifier on page 70
- Verifying the Forwarding Classes and Output Queues on page 72
- Verifying That the Classifier Is Applied to the Interfaces on page 72
- Verifying Behavior Aggregate Classifiers on page 73

Verifying the Code-Point Aliases

Purpose Make sure that the code-point aliases are configured as expected.

Action On Device R2, run the **show class-of-service code-point-aliases dscp** command.

```
user@R2> show class-of-service code-point-aliases dscp

Code point type: dscp
  Alias               Bit pattern
  af11                001010
  af12                001100
  af13                001110
```

```
af21                010010
af22                010100
af23                010110
af31                011010
af32                011100
af33                011110
af41                100010
af42                100100
af43                100110
be                  000000
be1                 000001
cs1                 001000
cs2                 010000
cs3                 011000
cs4                 100000
cs5                 101000
cs6                 110000
cs7                 111000
ef                  101110
ef1                 101111
nc1                 110000
nc2                 111000
```

Meaning The code-point aliases are configured as expected. Notice that the custom aliases that you configure are added to the default code-point aliases.

Verifying the DSCP Classifier

Purpose Make sure that the DSCP classifier is configured as expected.

Action On Device R2, run the **show class-of-service classifiers name v4-ba-classifier** command.

```
user@R2> show class-of-service classifiers name v4-ba-classifier

Classifier: v4-ba-classifier, Code point type: dscp, Index: 10755
  Code point         Forwarding class                    Loss priority
  000000             BE-data                             high
  000001             BE-data                             low
  000010             BE-data                             low
  000011             BE-data                             low
  000100             BE-data                             low
  000101             BE-data                             low
  000110             BE-data                             low
  000111             BE-data                             low
  001000             BE-data                             low
  001001             BE-data                             low
  001010             Voice                               low
  001011             BE-data                             low
  001100             Voice                               high
  001101             BE-data                             low
  001110             Voice                               high
  001111             BE-data                             low
```


Copyright © 2019, Juniper Networks, Inc.

```
010000              BE-data                         low
010001              BE-data                         low
010010              BE-data                         low
010011              BE-data                         low
010100              BE-data                         low
010101              BE-data                         low
010110              BE-data                         low
010111              BE-data                         low
011000              BE-data                         low
011001              BE-data                         low
011010              BE-data                         low
011011              BE-data                         low
011100              BE-data                         low
011101              BE-data                         low
011110              BE-data                         low
011111              BE-data                         low
100000              BE-data                         low
100001              BE-data                         low
100010              BE-data                         low
100011              BE-data                         low
100100              BE-data                         low
100101              BE-data                         low
100110              BE-data                         low
100111              BE-data                         low
101000              BE-data                         low
101001              BE-data                         low
101010              BE-data                         low
101011              BE-data                         low
101100              BE-data                         low
101101              BE-data                         low
101110              Premium-data                    high
101111              Premium-data                    low
110000              NC                              low
110001              BE-data                         low
110010              BE-data                         low
110011              BE-data                         low
110100              BE-data                         low
110101              BE-data                         low
110110              BE-data                         low
110111              BE-data                         low
111000              NC                              low
111001              BE-data                         low
111010              BE-data                         low
111011              BE-data                         low
111100              BE-data                         low
111101              BE-data                         low
111110              BE-data                         low
111111              BE-data                         low
```

Meaning Notice that the default classifier is incorporated into the customer classifier. If you were to remove the **import default** statement from the custom classifier, the custom classifier would look like this:

```
user@R2> show class-of-service classifier name v4-ba-classifier

Classifier: v4-ba-classifier, Code point type: dscp, Index: 10755
  Code point         Forwarding class                  Loss priority
```

```
000000              BE-data                         high
000001              BE-data                         low
101110              Premium-data                    high
101111              Premium-data                    low
```

Verifying the Forwarding Classes and Output Queues

Purpose Make sure that the forwarding classes are configured as expected.

Action On Device R2, run the **show class-of-service forwarding-class** command.

```
user@R2> show class-of-service forwarding-class


Forwarding class                          ID      Queue  Restricted queue  Fabric
priority  Policing priority   SPU priority
  BE-data                                  0       0          0             low
            normal             low
  Premium-data                             1       1          1             low
            normal             low
  Voice                                    2       2          2             low
            normal             low
  NC                                       3       3          3             low
            normal             low
```

Meaning The forwarding classes are configured as expected.

Verifying That the Classifier Is Applied to the Interfaces

Purpose Make sure that the classifier is applied to the correct interfaces.

Action On Device R2, run the **show class-of-service interface** command.

```
user@R2> show class-of-service interface ge-1/0/3


Physical interface: ge-1/0/3, Index: 144
Queues supported: 8, Queues in use: 4
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled

  Logical interface: ge-1/0/3.0, Index: 333
Object                  Name                    Type                    Index
Classifier              v4-ba-classifier        dscp                    10755
```

```
user@R2> show class-of-service interface ge-1/0/9


Physical interface: ge-1/0/9, Index: 150
Queues supported: 8, Queues in use: 4
```


Copyright © 2019, Juniper Networks, Inc.

```
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled

  Logical interface: ge-1/0/9.0, Index: 332
Object                  Name                  Type                  Index
Classifier              v4-ba-classifier      dscp                  10755
```

Meaning The interfaces are configured as expected.

Verifying Behavior Aggregate Classifiers

Purpose Verify that the behavior aggregate classifiers were configured properly on the device.

Action From configuration mode, enter the **show class-of-service** command.

When you are using hping to set the DSCP code points in the IPv4 packet header, the type-of-service (ToS) hex value (in this case, BC) is required in the **--tos** option of the **hping** command.

If your binary-to-hex or binary-to-decimal conversion skills are rusty, you can use an online calculator, such as http://www.mathsisfun.com/binary-decimal-hexadecimal-converter.html .

NOTE: When you convert a binary DSCP code point value, be sure to add two extra zeros at the end. So instead of 101111, use 10111100. These 0 values (the 7th and 8th bits) are reserved and ignored, but if you do not include them in the conversion, your hex and decimal values will be incorrect.

Extended Ping Sent from Device R1

```
user@R1> ping 172.16.70.1 tos 188 rapid count 25

PING 172.16.70.1 (172.16.70.1): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!
--- 172.16.70.1 ping statistics ---
25 packets transmitted, 25 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.404/0.483/1.395/0.207 ms
```

hping Sent from Host 1

```
root@host1> hping 172.16.70.1 --tos BC -c 25

HPING 172.16.70.1 (eth1 172.16.70.1): NO FLAGS are set, 40 headers + 0 data bytes
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=0.3 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=1 win=0 rtt=0.6 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=2 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=3 win=0 rtt=0.4 ms
```

Copyright © 2019, Juniper Networks, Inc.

```
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=4 win=0 rtt=0.6 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=5 win=0 rtt=0.3 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=6 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=7 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=8 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=9 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=10 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=11 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=12 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=13 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=14 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=15 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=16 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=17 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=18 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=19 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=20 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=21 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=22 win=0 rtt=0.4 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=23 win=0 rtt=0.5 ms
len=46 ip=172.16.70.1 ttl=61 DF id=0 sport=0 flags=RA seq=24 win=0 rtt=0.4 ms
```

On Device R2, Verify that Queue 2 is Incrementing.

Code point 101111 is associated with Premium-data, which uses queue 1.

```
user@R2> show interfaces extensive ge-1/0/3 | find "queue counters"
```

```
Queue counters:   Queued packets   Transmitted packets   Dropped packets
  0                          0                     0                 0
  1                         50                    50                 0
  2                          0                     0                 0
  3                         42                    42                 0
  Queue number:         Mapped forwarding classes
    0                   BE-data
    1                   Premium-data
    2                   Voice
    3                   NC
...
```

Meaning The output shows that queue 1 has incremented by 50 packets after sending 50 packets through the router.

Related Documentation

- *Interfaces Feature Guide for Security Devices*
- *Classification Overview*
- *Sample Behavior Aggregate Classification*
- *Understanding Packet Loss Priorities*

Copyright © 2019, Juniper Networks, Inc.

Understanding DSCP Classification for VPLS

You can perform Differentiated Services Code Point (DSCP) classification for IPv4 packets on Ethernet interfaces that are part of a virtual private LAN service (VPLS) routing instance on the ingress provider edge (PE) router. This is supported on the M320 router with Enhanced type III FPC and the M120 router. On the ATM II IQ PIC, the **ether-vpls-over-atm-llc** encapsulation statement is required. On the Intelligent Queuing 2 (IQ2) or Intelligent Queuing 2 Enhanced (IQ2E) PICs, the **vlan-vpls** encapsulation statement is required. DSCP for IPv6 and Internet precedence for IPv6 are not supported.

In order to perform DSCP classification for IPv4 packets on Ethernet interfaces that are part of a VPLS routing instance on the ingress PE router, you must make sure of the following:

- The correct encapsulation statement based on PIC type is configured for the interface.
- The DSCP classifier is defined (default is allowed) at the **[edit class-of-service classifiers]** hierarchy level.
- The defined DSCP classifier is applied to the interface.
- The interface is included in the VPLS routing instance on the ingress of the PE router.

A VPLS routing instance with the **no-tunnel-services** option configured has a default MPLS EXP classifier applied to the label-switched interface for all VPLS packets coming from the remote VPLS PE. This default classifier is modifiable only on MX Series routers. On T Series, when **no-tunnel-services** option is configured, the custom classifier for VPLS instances is not supported.

NOTE: With no-tunnel-services configured, a custom classifier for VPLS routing instances on T Series and LMNR based FPC for M320 is not supported. When a wild card configuration or explicit routing instances are configured for VPLS on CoS CLI, the custom classifier binding results in default classifier binding on Packet Forwarding Engine (PFE).

For example, on routing devices with eight queues (Juniper Networks M120 and M320 Multiservice Edge Routers, MX Series 5G Universal Routing Platforms, and T Series Core Routers), the default classification applied to **no-tunnel-services** VPLS packets are shown in Table 13 on page 75.

Table 13: Default VPLS Classifiers

MPLS Label EXP Bits	Forwarding Class/Queue
000	0
001	1
010	2

Copyright © 2019, Juniper Networks, Inc.

Table 13: Default VPLS Classifiers (continued)

MPLS Label EXP Bits	Forwarding Class/Queue
011	3
100	4
101	5
110	6
111	7

NOTE: Forwarding class to queue number mapping is not always one-to-one. Forwarding classes and queues are only the same when default forwarding-class-to-queue mapping is in effect. For more information about configuring forwarding class and queues, see "Configuring a Custom Forwarding Class for Each Queue" on page 212.

On MX Series routers, VPLS filters and policers act on a Layer 2 frame that includes the media access control (MAC) header (after any VLAN rewrite or other rules are applied), but does not include the cyclical redundancy check (CRC) field.

NOTE: On MX Series routers, if you apply a counter in a firewall for egress MPLS or VPLS packets with the EXP bits set to 0, the counter will not tally these packets.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36

Example: Configuring DSCP Classification for VPLS

This example shows how to configure a DSCP classifier for a virtual private LAN service (VPLS).

- Requirements on page 77
- Overview on page 77
- Configuration on page 77

Copyright © 2019, Juniper Networks, Inc.

Requirements

This example uses the following hardware and software components:

- An M Series Multiservice Edge Router (M120 and M320 only), MX Series 5G Universal Routing Platform, or T Series Core Router (TX Matrix and TX Matrix Plus only) with an ATM interface.
- Junos OS Release 10.4 or later.

Overview

In this example, you configure a DSCP classifier **dscp_vpls** on ATM interface **at-4/1/1** with **ether-vpls-over-atm-llc** encapsulation. The classifier **dscp_vpls** is applied to the interface and the interface is listed in the VPLS routing instance **vpls1** on the ingress PE router.

Configuration

CLI Quick Configuration

To quickly configure the DSCP classifier for a virtual private LAN service (VPLS), copy the following commands to a text file, remove any line breaks, and then paste the commands into the CLI.

```
user@host# set interfaces at-4/1/1 mtu 9192
user@host# set interfaces at-4/1/1 atm-options vpi 10
user@host# set interfaces at-4/1/1 unit 0 encapsulation ether-vpls-over-atm-llc
user@host# set interfaces at-4/1/1 unit 0 vci 10.128
user@host# set interfaces at-4/1/1 unit 0 family vpls
user@host# set class-of-service classifiers dscp dscp_vpls forwarding-class
  expedited-forwarding loss-priority low code-points 000010
user@host# set interfaces at-4/1/1 unit 0 classifiers dscp dscp_vpls
user@host# set routing-instances vpls1 instance-type vpls
user@host# set routing-instances vpls1 interface at-4/1/1.0
user@host# set routing-instances vpls1 route-distinguisher 10.255.245.51:1
user@host# set routing-instances vpls1 vrf-target target:1234:1
user@host# set routing-instances vpls1 protocols vpls site-range 10
user@host# set routing-instances vpls1 protocols vpls no-tunnel-services
user@host# set routing-instances vpls1 protocols vpls site vpls-1-site-1 site-identifier 1
```

Configuring the DSCP Classifier for a Virtual Private LAN Service (VPLS)

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the DSCP classifier for a virtual private LAN service (VPLS:

1. Configure the ATM interface **at-4/1/1.0** and the encapsulation as **ether-vpls-over-atm-llc**.

   ```
   [edit interfaces]
   user@host# set at-4/1/1 mtu 9192
   ```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set at-4/1/1 atm-options vpi 10
user@host# set at-4/1/1 unit 0 encapsulation ether-vpls-over-atm-llc
user@host# set at-4/1/1 unit 0 vci 10.128
user@host# set at-4/1/1 unit 0 family vpls
```

2. Configure the DSCP classifier **dscp_vpls**.

```
[edit class-of-service]
user@host# set classifiers dscp dscp_vpls forwarding-class expedited-forwarding
  loss-priority low code-points 000010
```

3. Apply the classifier **dscp_vpls** to the ATM interface **at-4/1/1.0**.

```
[edit interfaces]
user@host# set at-4/1/1 unit 0 classifiers dscp dscp_vpls
```

4. Include the ATM interface virtual circuit **at-4/1/1.0** as part of the routing instance **vpls1** configuration.

```
user@host# set routing-instances vpls1 instance-type vpls
user@host# set routing-instances vpls1 interface at-4/1/1.0
user@host# set routing-instances vpls1 route-distinguisher 10.255.245.51:1
user@host# set routing-instances vpls1 vrf-target target:1234:1
user@host# set routing-instances vpls1 protocols vpls site-range 10
user@host# set routing-instances vpls1 protocols vpls no-tunnel-services
user@host# set routing-instances vpls1 protocols vpls site vpls-1-site-1 site-identifier
  1
```

Results

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

```
user@host# show interfaces at-4/1/1
```

```
mtu 9192;
atm-options {
    vpi 10;
}
unit 0 {
     classifiers {
       dscp dscp_vpls;
}
     encapsulation ether-vpls-over-atm-llc;
     vci 10.128;
```


Copyright © 2019, Juniper Networks, Inc.

```
    family vpls;
}
```

```
user@host# show class-of-service
```

```
classifiers {
    dscp dscp_vpls {
        forwarding-class expedited-forwarding {
            loss-priority low code-points 000010;
        }
    }
}
```

```
user@host# show routing-instances
```

```
vpls1 {
    instance-type vpls;
    interface at-4/1/1.0;
    route-distinguisher 10.255.245.51:1;
    vrf-target target:1234:1;
    protocols {
        vpls {
            site-range 10;
            no-tunnel-services;
            site vpls-1-site-1 {
                site-identifier 1;
            }
        }
    }
}
```

Related Documentation

- Understanding DSCP Classification for VPLS on page 75

Configuring Class of Service for MPLS LSPs

The following sections provide an overview of MPLS class of service (CoS) and describe how to configure the MPLS CoS value:

- Class of Service for MPLS Overview on page 79
- Configuring the MPLS CoS Values on page 80
- Rewriting IEEE 802.1p Packet Headers with the MPLS CoS Value on page 82

Class of Service for MPLS Overview

When IP traffic enters an LSP tunnel, the ingress router marks all packets with a CoS value, which is used to place the traffic into a transmission priority queue. On the router, for SDH/SONET and T3 interfaces, each interface has four transmit queues. The CoS

Copyright © 2019, Juniper Networks, Inc.

value is encoded as part of the MPLS header and remains in the packets until the MPLS header is removed when the packets exit from the egress router. The routers within the LSP utilize the CoS value set at the ingress router. The CoS value is encoded by means of the CoS bits (also known as the EXP or experimental bits). For more information, see *MPLS Label Allocation*.

MPLS class of service works in conjunction with the router's general CoS functionality. If you do not configure any CoS features, the default general CoS settings are used. For MPLS class of service, you might want to prioritize how the transmit queues are serviced by configuring weighted round-robin, and to configure congestion avoidance using random early detection (RED)..

Configuring the MPLS CoS Values

When traffic enters an LSP tunnel, the CoS value in the MPLS header is set in one of three ways:

- The number of the output queue into which the packet was buffered and the packet loss priority (PLP) bit are written into the MPLS header and are used as the packet's CoS value. This behavior is the default, and no configuration is required. "Default MPLS EXP Classifier" on page 42 explains the default MPLS CoS values, and summarizes how the CoS values are treated.
- You set a fixed CoS value on all packets entering the LSP tunnel. A fixed CoS value means that all packets entering the LSP receive the same class of service.
- You set an MPLS EXP rewrite rule to override the default behavior.

To set a fixed CoS value on all packets entering the LSP, include the **class-of-service** statement:

```
class-of-service cos-value;
```

You can include this statement at the following hierarchy levels:

- **[edit protocols mpls]**
- **[edit protocols mpls label-switched-path *path-name*]**
- **[edit protocols mpls label-switched-path *path-name* primary *path-name*]**
- **[edit protocols mpls label-switched-path *path-name* secondary *path-name*]**
- **[edit protocols rsvp interface *interface-name* link-protection]**
- **[edit protocols rsvp interface *interface-name* link-protection bypass *destination*]**
- **[edit logical-systems *logical-system-name* protocols mpls]**
- **[edit logical-systems *logical-system-name* protocols mpls label-switched-path *path-name*]**
- **[edit logical-systems *logical-system-name* protocols mpls label-switched-path *path-name* primary *path-name*]**

Copyright © 2019, Juniper Networks, Inc.

- **[edit logical-systems *logical-system-name* protocols mpls label-switched-path *path-name* secondary *path-name*]**
- **[edit logical-systems *logical-system-name* protocols rsvp interface *interface-name* link-protection]**
- **[edit logical-systems *logical-system-name* protocols rsvp interface *interface-name* link-protection bypass *destination*]**

The CoS value set using the **class-of-service** statement at the **[edit protocols mpls]** hierarchy level supersedes the CoS value set at the **[edit class-of-service]** hierarchy level for an interface. Effectively, the CoS value configured for an LSP overrides the CoS value set for an interface.

The **class-of-service** statement at the **[edit protocols mpls label-switched-path]** hierarchy level assigns an initial EXP value for the MPLS shim header of packets in the LSP. This value is initialized at the ingress routing device only and overrides the rewrite configuration established for that forwarding class. However, the CoS processing (weighted round robin [WRR] and RED) of packets entering the ingress routing device is not changed by the **class-of-service** statement on an MPLS LSP. Classification is still based on the behavior aggregate (BA) classifier at the **[edit class-of-service]** hierarchy level or the multifield classifier at the **[edit firewall]** hierarchy level.

BEST PRACTICE: We recommend configuring all routing devices along the LSP to have the same input classifier for EXP, and, if a rewrite rule is configured, all routing devices should have the same rewrite configuration. Otherwise, traffic at the next LSR might be classified into a different forwarding class, resulting in a different EXP value being written to the EXP header.

The CoS value can be a decimal number from 0 through 7. This number corresponds to a 3-bit binary number. The high-order 2 bits of the CoS value select which transmit queue to use on the outbound interface card.

The low-order bit of the CoS value is treated as the PLP bit and is used to select the RED drop profile to use on the output queue. If the low-order bit is 0, the non-PLP drop profile is used, and if the low-order bit is 1, the PLP drop profile is used. It is generally expected that RED will more aggressively drop packets that have the PLP bit set. For more information about RED and drop profiles, see "Managing Congestion Using RED Drop Profiles and Packet Loss Priorities" on page 351.

NOTE: Configuring the PLP drop profile to drop packets more aggressively (for example, setting the CoS value from 6 to 7) decreases the likelihood of traffic getting through.

Table 14 on page 82 summarizes how MPLS CoS values correspond to the transmit queue and PLP bit. Note that in MPLS, the mapping between the CoS bit value and the output queue is hard-coded. You cannot configure the mapping for MPLS; you can configure it

Copyright © 2019, Juniper Networks, Inc.

only for IPv4 traffic flows, as described in "Understanding How Forwarding Classes Assign Classes to Output Queues" on page 205.

Table 14: MPLS CoS Values

MPLS CoS Value	Bits	Transmit Queue	PLP Bit
0	000	0	Not set
1	001	0	Set
2	010	1	Not set
3	011	1	Set
4	100	2	Not set
5	101	2	Set
6	110	3	Not set
7	111	3	Set

Because the CoS value is part of the MPLS header, the value is associated with the packets only as they travel through the LSP tunnel. The value is not copied back to the IP header when the packets exit from the LSP tunnel.

To configure class of service (CoS) for Multiprotocol Label Switching (MPLS) packets in a label-switched path (LSP):

1. Specify the CoS value

   If you do not specify a CoS value, the IP precedence bits from the packet's IP header are used as the packet's CoS value.

Rewriting IEEE 802.1p Packet Headers with the MPLS CoS Value

For Ethernet interfaces installed on a T Series router or an M320 router with a peer connection to an M Series router or a T Series router, you can rewrite both MPLS CoS and IEEE 802.1p values to a configured value (the MPLS CoS values are also known as the EXP or experimental bits). Rewriting these values allows you to pass the configured value to the Layer 2 VLAN path. To rewrite both the MPLS CoS and IEEE 802.1p values, you must include the EXP and IEEE 802.1p rewrite rules in the class of service interface configuration. The EXP rewrite table is applied when you configure the IEEE 802.1p and EXP rewrite rules.

For information about how to configure the EXP and IEEE 802.1p rewrite rules, see "Rewriting Packet Headers to Ensure Forwarding Behavior" on page 388.

Related Documentation

- Default MPLS EXP Classifier on page 42

Copyright © 2019, Juniper Networks, Inc.

Applying DSCP Classifiers to MPLS Traffic

On MX960, MX480, MX240, MX80, M120, and M320 routers with Enhanced Type III FPCs and EX Series switches only, you can configure user-defined DSCP-based BA classification for MPLS interfaces or VPLS/L3VPN routing instances (LSI interfaces).

NOTE: You cannot configure user-defined DSCP-based BA classification for MPLS interfaces on IQE PICs or on MX Series routers or EX Series switches when ingress queuing is used.

The following examples show how you can apply DSCP classifiers for MPLS traffic on core-facing interfaces and VPLS/L3VPN routing instances. These classifiers are applicable on egress PE routers for VPLS and L3VPN cases. For plain interfaces (not VPLS/L3VPN (LSI) interfaces), these classifiers are applicable on P and egress PE routers on core-facing interfaces.

- Applying a DSCP Classifier to MPLS Packets on a Core-facing Interface on page 83
- Applying a DSCP Classifier to MPLS Traffic for L3VPN/VPLS on page 85

Applying a DSCP Classifier to MPLS Packets on a Core-facing Interface

The following procedure requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

The following example:

a. Configures core-facing interface ge-5/3/1.0 for protocol families IPv4, IPv6, and International Organization for Standardization Open Systems Interconnection (ISO OSI)

b. Configures the DSCP classifier **dscp11**.

c. Apply the DSCP classifier to the logical interface for the MPLS family.

To configure and apply a DSCP classifier to MPLS packets on a core-facing interface:

1. Configure the core-facing interface and associated logical interfaces.

```
[edit interfaces ge-5/3/1 unit 0]
user@host # set family inet address 10.1.1.1/24
user@host # set family iso
user@host # set family inet6 address 2001:db8::1/64
user@host # set family mpls
```

2. Configure the DSCP classifier.

```
[edit class-of-service classifiers dscp dscp11]
```

Copyright © 2019, Juniper Networks, Inc.

```
user@host # set forwarding-class expedited-forwarding loss-priority low code-points
  [ef cs5]
user@host # set forwarding-class assured-forwarding loss-priority low code-points
  [af21 af31 af41 cs4]
user@host # set forwarding-class assured-forwarding loss-priority high code-points
  [af23 af33 af43 cs2 af22 af32 af42 cs3]
user@host # set forwarding-class best-effort loss-priority low code-points [af11 cs1
  af12]
user@host # set forwarding-class best-effort loss-priority high code-points af13
user@host # set forwarding-class network-control loss-priority low code-points [cs6
  cs7]
```

3. Apply the classifier to the logical interface for the MPLS family.

NOTE: You cannot configure more than one classifier per family.

```
[edit class-of-service interfaces ge-5/3/1 unit 0]
user@host # set classifiers dscp dscp11 family mpls
```

4. Confirm the configuration.

```
[edit interfaces ge-5/3/1 unit 0]
user@host# show
```

```
family inet {
    address 10.1.1.1/24;
}
family iso;
family inet6 {
    address 2001:db8::1/64;
}
family mpls;
```

```
[edit class-of-service classifiers dscp dscp11]
user@host# show
```

```
forwarding-class expedited-forwarding {
    loss-priority low code-points [ ef cs5 ];
}
forwarding-class assured-forwarding {
    loss-priority low code-points [ af21 af31 af41 cs4 ];
    loss-priority high code-points [ af23 af33 af43 cs2 af22 af32 af42 cs3 ];
}
forwarding-class best-effort {
    loss-priority low code-points [ af11 cs1 af12 ];
    loss-priority high code-points af13;
}
forwarding-class network-control {
    loss-priority low code-points [ cs6 cs7 ];
```


Copyright © 2019, Juniper Networks, Inc.

```
}
```

```
[edit class-of-service interfaces ge-5/3/1 unit 0]
user@host# show
```

```
classifiers {
    dscp dscp11 {
        family mpls;
    }
}
```

5. Save the configuration.

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

Applying a DSCP Classifier to MPLS Traffic for L3VPN/VPLS

The following procedure requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

The following example:

a. Configures routing instances of type either vrf or vpls.

b. Configures the DSCP classifier.

c. Attaches the classifier to the routing instance.

To configure and apply a DSCP classifier to MPLS traffic for L3VPN/VPLS:

1. Configure routing instances of type either vrf or vpls.

```
[edit routing-instances vpls1]
user@host# set instance-type vpls
user@host# set interface ge-2/2/2.0
user@host# set route-distinguisher 10.255.245.51:1
user@host# set vrf-target target:1234:1
user@host# set protocols vpls site-range 10
user@host# set protocols vpls no-tunnel-services
user@host# set protocols vpls site vpls-1-site-1 site-identifier 1
```

2. Configure the DSCP classifier.

```
[edit class-of-service classifiers dscp dscp11]
user@host # set forwarding-class expedited-forwarding loss-priority low code-points
  [ef cs5]
```

Copyright © 2019, Juniper Networks, Inc.

```
user@host # set forwarding-class assured-forwarding loss-priority low code-points
  [af21 af31 af41 cs4]
user@host # set forwarding-class assured-forwarding loss-priority high code-points
  [af23 af33 af43 cs2 af22 af32 af42 cs3]
user@host # set forwarding-class best-effort loss-priority low code-points [af11 cs1
  af12]
user@host # set forwarding-class best-effort loss-priority high code-points af13
user@host # set forwarding-class network-control loss-priority low code-points [cs6
  cs7]
```

3. Attach the classifier to the routing instance.

```
[edit class-of-service routing-instances vpls1]
user@host # set classifiers dscp dscp11
```

NOTE: You cannot configure more than one classifier per routing instance.

4. Confirm the configuration.

```
[edit routing-instances vpls1]
user@host# show
```

```
instance-type vpls;
interface ge-2/2/2.0; ## customer facing interface
route-distinguisher 10.255.245.51:1;
vrf-target target:1234:1;
protocols {
    vpls {
        site-range 10;
        no-tunnel-services;
        site vpls-1-site-1 {
            site-identifier 1;
        }
    }
}
```

```
[edit class-of-service]
user@host# show
```

```
classifiers {
    dscp dscp11 {
        forwarding-class expedited-forwarding {
            loss-priority low code-points [ ef cs5 ];
        }
        forwarding-class assured-forwarding {
            loss-priority low code-points [ af21 af31 af41 cs4 ];
           loss-priority high code-points [ af23 af33 af43 cs2 af22 af32 af42
 cs3 ];
        }
```


Copyright © 2019, Juniper Networks, Inc.

```
        forwarding-class best-effort {
            loss-priority low code-points [ af11 cs1 af12 ];
            loss-priority high code-points af13;
        }
        forwarding-class network-control {
            loss-priority low code-points [ cs6 cs7 ];
        }
    }
}
routing-instances {
    vpls1 {
        classifiers {
            dscp dscp11;
        }
    }
}
```

5. Save the configuration.

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

Related Documentation

- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53

Applying MPLS EXP Classifiers to Routing Instances

This topic shows how to apply MPLS EXP classifiers to routing instances.

When you enable VRF table labels and you do not explicitly apply a classifier configuration to the routing instance, the default MPLS EXP classifier is applied to the routing instance. For detailed information about VRF table labels, see the *Junos OS VPNs Library for Routing Devices*.

The default MPLS EXP classification table contents are shown in Table 15 on page 87.

Table 15: Default MPLS EXP Classifier

MPLS EXP Bits	Forwarding Class	Loss Priority
000	best-effort	low
001	best-effort	high
010	expedited-forwarding	low
011	expedited-forwarding	high
100	assured-forwarding	low
101	assured-forwarding	high

Copyright © 2019, Juniper Networks, Inc.

Table 15: Default MPLS EXP Classifier (continued)

MPLS EXP Bits	Forwarding Class	Loss Priority
110	network-control	low
111	network-control	high

NOTE: At times you might need to maintain the original classifier—for example with bridge domains, where you neither want to configure a custom classifier for the routing instance nor accept the default classifier, which would override the original classifier. Starting with Junos OS Release 16.1, on MX Series devices only, you can maintain the original MPLS EXP classifier. To do so, apply the **no-default** option for the routing instance. For example:

```
[edit class-of-service]
routing-instances routing-instance-name {
  classifiers {
    no-default;
  }
}
```

This topic describes:

- Configuring and Applying Custom MPLS EXP Classifiers to Routing Instances on page 88
- Applying Global Classifiers and Wildcard Routing Instances on page 89
- Applying Global MPLS EXP Classifiers to Routing Instances on page 90
- Applying Classifiers by Using Wildcard Routing Instances on page 91
- Verifying the Classifiers Associated with Routing Instances on page 93

Configuring and Applying Custom MPLS EXP Classifiers to Routing Instances

NOTE: The following caveats apply to custom MPLS EXP classifiers for routing instances:

- An enhanced FPC is required.
- Logical systems are not supported.

Copyright © 2019, Juniper Networks, Inc.

For PICs that are installed on enhanced FPCs, you can override the default MPLS EXP classifier and apply a custom classifier to a routing instance.

The following procedure requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To apply a custom classifier to a routing instance:

1. Filter traffic based on the IP header.

```
[edit]
user@host# edit routing-instances routing-instance-name
user@host# set vrf-table-label
```

2. Configure the custom MPLS EXP classifier.

```
[edit]
user@host# edit class-of-service
user@host# set classifiers exp classifier-name import classifier-name forwarding-class
   class-name loss-priority level code-points [ aliases ] [ bit-patterns]
user@host# set forwarding-classes queue queue-number class-name priority (high |
   low)
```

3. Apply the custom MPLS EXP classifier to the routing instance..

```
[edit class-of-service routing-instances routing-instance-name classifiers]
user@host# set exp classifier-name;
```

4. Commit and confirm your configuration.

```
[edit]
user@host# show class-of-service routing-instances
```

Applying Global Classifiers and Wildcard Routing Instances

To apply a classifier to all routing instances:

- Specify that the MPLS EXP classifier is for all routing instances.

```
[edit class-of-service ]
user@host# set routing-instances all classifiers exp classifier-name
```

For routing instances associated with specific classifiers, the global configuration is ignored.

Copyright © 2019, Juniper Networks, Inc.

To use a wildcard to apply a classifier to all routing instances:

- Include an asterisk (*) in the name of the routing instance.

```
[edit]]
user@host# edit class-of-service routing-instances routing-instance-name*
user@host# set classifiers exp classifier-name
```

The wildcard configuration follows the longest match. If there is a specific configuration, it is given precedence over the wildcard configuration.

NOTE: The wildcard * and the **all** keyword are supported at the **[edit class-of-service routing-instances]** hierarchy level but not at the **[edit routing-instances]** hierarchy level.

If you configure a routing instance at the **[edit routing-instances]** hierarchy level with, for example, the name **vpn***, Junos OS treats **vpn*** as a valid and distinct routing instance name. If you then try to apply a classifier to the **vpn*** routing instance at the **[edit class-of-service routing-instances]** hierarchy level, the Junos OS treats the **vpn*** routing instance name as a wildcard, and all routing instances that start with **vpn** and do not have a specific classifier applied receive the classifier associated with **vpn***.

This same behavior applies with the **all** keyword.

Note that the * wildcard *must* be appended to an instance name at these configuration levels. The * wildcard should not be intended as a stand-alone substitute for the **all** keyword.

Applying Global MPLS EXP Classifiers to Routing Instances

This example shows how to apply a global classifier to all routing instances and then override the global classifier for a specific routing instance. In this example, there are three routing instances: **vpn1**, **vpn2**, and **vpn3**, each with VRF table label enabled. The classifier **exp-classifier-global** is applied to **vpn1** and **vpn2** (that is, all but **vpn3**, which is listed separately). The classifier **exp-classifier-3** is applied to **vpn3**.

The following procedure requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a global classifier for all routing instances and override the global classifier for a specific routing instance:

1. Enable the VRF table label for all three routing instances.

```
[edit routing-instances]
user@host# set vpn1 vrf-table-label
user@host# set vpn2 vrf-table-label
user@host# set vpn3 vrf-table-label
```


Copyright © 2019, Juniper Networks, Inc.

2. Apply the EXP classifier **exp-classifier-global** to all routing instances.

```
[edit class-of-service routing-instances]
user@host# set all classifiers exp exp-classifier-global
```

3. Apply the EXP classifier **exp-classifier-3** to only the routing-instance **vpn3**.

```
[edit class-of-service routing-instances]
user@host# set vpn3 classifiers exp exp-classifier-3
```

4. Confirm your configuration.

```
[edit routing-instances]
user@host# show
```

```
vpn1 {
vrf-table-label;
}
vpn2 {
vrf-table-label;
}
vpn3 {
vrf-table-label;
}
[edit class-of-service routing-instances]
```

```
[edit class-of-service routing-instances]
user@host# show
```

```
all {
classifiers {
exp exp-classifier-global;
}
}
vpn3 {
classifiers {
exp exp-classifier-3;
}
}
```

Applying Classifiers by Using Wildcard Routing Instances

Configure a wildcard routing instance and override the wildcard with a specific routing instance. In this example, there are three routing instances: **vpn-red**, **vpn-yellow**, and **vpn-green**, each with VRF table label enabled. The classifier **exp-class-wildcard** is applied to **vpn-yellow** and **vpn-green**. The classifier **exp-class-red** is applied to **vpn-red**.

Copyright © 2019, Juniper Networks, Inc.

The following procedure requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure a wildcard routing instance and override the wildcard with a specific routing instance:

1. Enable the VRF table label for all three routing instances.

```
[edit routing-instances]
user@host# set vpn-red vrf-table-label
user@host# set vpn-yellow vrf-table-label
user@host# set vpn-green vrf-table-label
```

2. Apply the EXP classifier **exp-class-wildcard** to all routing instances by using a wildcard.

```
[edit class-of-service routing-instances]
user@host# set vpn* classifiers exp exp-class-wildcard
```

3. Apply the EXP classifier **exp-class-red** to only the routing-instance **vpn-red**.

```
[edit class-of-service routing-instances]
user@host# set vpn-red classifiers exp exp-class-red
```

4. Commit and confirm your configuration.

```
[edit routing-instances]
user@host# show
```

```
vpn-red {
vrf-table-label;
}
vpn-yellow {
vrf-table-label;
}
vpn-green {
vrf-table-label;
}
```

```
[edit class-of-service routing-instances]
user@host# show
```

```
vpn* {
classifiers {
exp exp-class-wildcard;
}
}
vpn-red {
classifiers {
exp exp-class-red;
```


Copyright © 2019, Juniper Networks, Inc.

```
}
}
```

Verifying the Classifiers Associated with Routing Instances

Purpose Display the MPLS EXP classifiers associated with two routing instances:

Action To verify the MPLS EXP classifiers associated with two routing instances, enter the following Junos OS CLI operational mode command:

```
user@host> show class-of-service routing-instances
```

```
Routing Instance : vpn1
  Object            Name                 Type                 Index
  Classifier        exp-default          exp                      8

Routing Instance : vpn2
  Object            Name                 Type                 Index
  Classifier        class2               exp                  57507
```

Release History Table

Release	Description
16.1	Starting with Junos OS Release 16.1, on MX Series devices only, you can maintain the original MPLS EXP classifier.

Related Documentation

- Configuring Behavior Aggregate Classifiers on page 51
- Default MPLS EXP Classifier on page 42
- Applying MPLS EXP Classifiers for Explicit-Null Labels on page 94

Copyright © 2019, Juniper Networks, Inc.

Applying MPLS EXP Classifiers for Explicit-Null Labels

When you configure MPLS explicit-null labels, label 0 is advertised to the egress router of an LSP. When label 0 is advertised, the egress router (instead of the penultimate router) removes the label. Ultimate-hop popping ensures that any packets traversing an MPLS network include a label. For more information about explicit-null labels and ultimate-hop popping, see the *MPLS Applications Feature Guide*.

On M320 and T Series routers, when you configure MPLS explicit-null labels with an MPLS EXP classifier, the MPLS EXP classifier can be different from an IPv4 or IPv6 classifier configured on the same logical interface. In other words, you can apply separate classifiers for MPLS EXP, IPv4, and IPv6 packets per logical interface. To combine an EXP classifier with a distinct IPv6 classifier, the PIC must be mounted on an Enhanced FPC.

NOTE: For M Series routers, MPLS explicit-null labels with MPLS EXP classification are supported if you set the same classifier for EXP and IPv4 traffic, or EXP and IPv6 traffic.

For more information about how IPv4 and IPv6 packet classification is handled, see "Applying Behavior Aggregate Classifiers to Logical Interfaces" on page 53.

To configure an MPLS EXP classifier for explicit-null labels:

1. Create the MPLS EXP classifier.

```
[edit]
user@host# edit class-of-service classifiers exp classifier-name
```

2. Specify the name of a predefined classifier to include in this configuration.

```
[edit class-of-service classifiers exp classifier-name]
user@host# set import classifier-name
```

3. Define a classification of code point aliases for the classifier.

```
[edit class-of-service classifiers exp classifier-name]
user@host# set forwarding-class class-name loss-priority level code-points value
```

To apply the MPLS EXP classifier to the logical interface:

1. Specify the physical and logical interface names on which you want to apply the classifier.

```
[edit]
```


Copyright © 2019, Juniper Networks, Inc.

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

2. Specify the classifier type and name you want to apply to the interface.

```
[edit class-of-service classifiers interfaces interface-name ]
user@host# set classifiers exp classifier-name
```

NOTE: When a packet with a single label is received, if the label is an explicit-null label (0 or 2), the label is popped first, making the EXP information no longer available. The subsequent packet classification is based on the IPv4/IPv6 payload. Starting with Junos OS 18.1R1, PTX Series routers with third-generation FPCs (FPC3) support a new CLI option, [explicit-null-cos *inet|inet6*] at the [edit forwarding-options] hierarchy level, that makes the packet classification based on the MPLS EXP value rather than on the payload, thus preserving the MPLS classification of the packet.

Release History Table

Release	Description
18.1	Starting with Junos OS 18.1R1, PTX Series routers with third-generation FPCs (FPC3) support a new CLI option, **[explicit-null-cos *inet|inet6*]** at the **[edit forwarding-options]** hierarchy level, that makes the packet classification based on the MPLS EXP value rather than on the payload, thus preserving the MPLS classification of the packet.

Related Documentation

- Configuring Behavior Aggregate Classifiers on page 51
- Default MPLS EXP Classifier on page 42
- Applying MPLS EXP Classifiers to Routing Instances on page 87

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 3

Assigning Service Levels with Multifield Classifiers

- Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 97
- Configuring Multifield Classifiers on page 98
- Using Multifield Classifiers to Set Packet Loss Priority on page 101
- Example: Configuring and Applying a Firewall Filter for a Multifield Classifier on page 103
- Example: Classifying Packets Based on Their Destination Address on page 109
- Example: Configuring and Verifying a Complex Multifield Filter on page 111

Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields

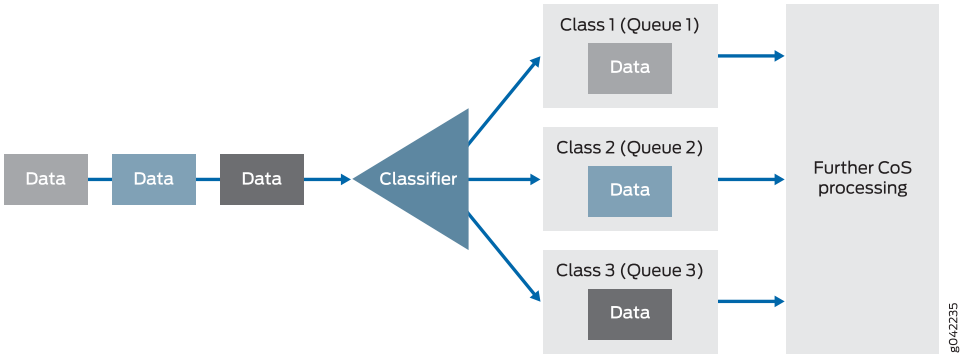
Behavior aggregate (BA) classification (see "Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic" on page 36), where packets are classified based on their QoS markings, is the most common way to assign service levels because it is straightforward and based on a well-established, fixed-length header fields, which makes them computationally more efficient. However, sometimes BA classification does not provide sufficient granularity, or the QoS markings in the packet headers cannot be trusted. In such situations, multifield classifiers can be used. A multifield classifier is a method of classifying traffic flows based on multiple packet header fields. Devices that sit at the edge of a network usually classify packets based on multiple packet header fields. Multifield classification is normally performed at the network edge because of the general lack of DiffServ code point (DSCP) or IP precedence support in end-user applications.

In an edge router, a multifield classifier provides the filtering functionality that scans through a variety of packet header fields to determine the forwarding class for a packet. Typically, a classifier performs matching operations on the selected fields against a configured value. A multifield classifier can examine multiple fields in the packet header: destination address, source address, IP protocol, source port, destination port, and DSCP value. Multifield classifiers are used when a simple BA classifier is insufficient to classify a packet.

Figure 13 on page 98 provides a high-level illustration of how a classifier works.

Copyright © 2019, Juniper Networks, Inc.

Figure 13: How a Classifier Works



In Junos OS, you configure a multifield classifier with a firewall filter and its associated match conditions. This enables you to use any filter match criteria to locate packets that require classification. From a CoS perspective, multifield classifiers (or firewall filter rules) provide the following services:

- Classify packets to a forwarding class and loss priority. The forwarding class determines the output queue. The loss priority is used by schedulers in conjunction with the random early discard (RED) algorithm to control packet discard during periods of congestion.
- Police traffic to a specific bandwidth and burst size. Packets exceeding the policer limits can be discarded, or can be assigned to a different forwarding class, to a different loss priority, or to both.

NOTE: You *police* traffic on input to conform to established CoS parameters, setting loss handling and forwarding class assignments as needed. You *shape* traffic on output to make sure that router resources, especially bandwidth, are distributed fairly. However, input policing and output shaping are two different CoS processes, each with their own configuration statements.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Configuring Multifield Classifiers on page 98

Configuring Multifield Classifiers

This topic describes how you configure multifield classifiers.

Multifield classifiers classify packets to a forwarding class and loss priority based on the filter match criteria. Multifield classification is usually done at the edge of the network for packets that do not have valid or trusted behavior aggregate code points.

If you configure both a behavior aggregate (BA) classifier and a multifield classifier, BA classification is performed first; then multifield classification is performed. If they conflict, any BA classification result is overridden by the multifield classifier.

Copyright © 2019, Juniper Networks, Inc.

NOTE: For a specified interface, you can configure both a multifield classifier and a BA classifier without conflicts. Because the classifiers are always applied in sequential order, the BA classifier followed by the multifield classifier, any BA classification result is overridden by a multifield classifier if they conflict.

To activate (apply) a multifield classifier, you must configure it on a logical interface. There is no restriction on the number of multifield classifiers you can configure.

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

For an L2TP LNS on MX Series routers, you can attach firewall for static LNS sessions by configuring these at logical interfaces directly on the inline services device (si-fpc/pic/port). RADIUS-configured firewall attachments are not supported.

You configure multifield classifiers by:

1. **Defining the filter**—Configure *either* a firewall filter or a simple filter. Simple filters filter IPv4 traffic (family inet) only. Firewall filters enable you to filter additional protocol families and more complex filters. The following sections describe both procedures.
2. **Applying the filter**—Activate the filter by configuring on a logical interface as an *input* filter.

To configure a firewall filter:

1. Under the **firewall** statement, specify the protocol family for which you want to filter traffic and specify a name for the filter.

```
edit
user@host# edit firewall family family-name filter filter-name
```

2. Specify the term name and match criteria you want to look for in incoming packets.

```
[edit firewall family family-name filter filter-name]
user@host# set term term-name from match-conditions
```

3. Specify the action you want to take when a packet matches the conditions.

```
[edit firewall family family-name filter filter-name]
user@host# set term term-name then actions
```

For multifield classifiers, you can perform the following actions:

- Set the value of the DSCP field of incoming packets.

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set term term-name then dscp code-point
```

- Set the forwarding class of incoming packets. The forwarding class determines the output queue.

```
user@host# set term term-name then forwarding-class class-name
```

- Set the loss priority of incoming packets. The loss priority is used by schedulers in conjunction with the random early discard (RED) algorithm to control packet discard during periods of congestion.

```
user@host# set term term-name then loss-priority (high | low | medium-high |
  medium-low)
```

To configure a simple filter:

1. Specify a name for the simple filter.

```
[edit firewall family family-name]
user@host# edit simple-filter filter-name
```

2. Specify the term name and match criteria you want to look for in incoming packets.

```
[edit firewall family family-name simple-filter filter-name]
user@host# set term term-name from match-conditions
```

3. Specify the action you want to take when a packet matches the conditions.

```
[edit firewall family family-name simple-filter filter-name]
user@host# set term term-name then actions
```

For multifield classifiers, you can perform the following actions for a simple filter:

- Set the forwarding-class of incoming packets.
- Set theloss-priority of incoming packets.

To apply the firewall filter to the appropriate logical interfaces as an input filter.

1. Specify the physical and logical interface on which you want to apply the firewall filter.

```
edit
user@host# edit interfaces interface-name unit unit-number
```

2. Specify the protocol family for the firewall filter.

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

3. Specify the names of the firewall filters to apply to received packets.

Copyright © 2019, Juniper Networks, Inc.

```
[edit interfaces interface-name unit unit-number]
user@host# set filter input filter-name
```

Repeat this step for the family protocol filter and the simple filter.

4. Save your configuration.

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

Related Documentation

- Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 97
- Configuring a Simple Filter on page 839
- *Guidelines for Applying Standard Firewall Filters*
- Using Multifield Classifiers to Set Packet Loss Priority on page 101

Using Multifield Classifiers to Set Packet Loss Priority

This topic describes how to use and configure multifield classifiers to set the loss priority of incoming or outgoing packets.

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

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

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

Copyright © 2019, Juniper Networks, Inc.

In the following sample procedure, the forwarding class **expedited-forwarding** and PLP **medium-high** are assigned to all IPv4 packets with the **10.1.1.0/24** or **10.1.2.0/24** source address.

To use the classifier in this sample procedure, you must configure the settings for the **expedited-forwarding** forwarding class at the **[edit class-of-service forwarding-classes queue *queue-number* expedited-forwarding]** hierarchy level. For more information, see "Understanding How Forwarding Classes Assign Classes to Output Queues" on page 205.

1. Under the **firewall** statement, specify the protocol family as IPv4 (inet) and specify a name for the filter.

```
edit
user@host# edit firewall family inet filter classify-customers
```

2. Specify the term name and match criteria you want to look for in incoming packets.

```
[edit firewall family inet filter classify-customers]
user@host# set term isp1-customers from source-address 10.1.1.0/24
user@host# set term isp1-customers from source-address 10.1.2.0/24
```

3. Specify the action you want to take when a packet matches the conditions.

```
[edit firewall family inet filter classify-customers]
user@host# set term isp1-customers then loss-priority medium-high
user@host# set term isp1-customers then forwarding-class medium-high
```

4. Verify your configuration.

```
[edit firewall]
user@host# show
```

```
filter classify-customers {
       term isp1-customers {
           from {
               source-address {
                   10.1.1.0/24;
                   10.1.2.0/24;
               }
           }
           then {
               loss-priority medium-low;
               forwarding-class assured-forwarding;
           }
       }
   }
```

5. Save your configuration.

```
[edit firewall]
user@host# commit
```


Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring Multifield Classifiers on page 98
- Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 97

Example: Configuring and Applying a Firewall Filter for a Multifield Classifier

This example shows how to configure a firewall filter to classify traffic using a multifield classifier. The classifier detects packets of interest to class of service (CoS) as they arrive on an interface. Multifield classifiers are used when a simple behavior aggregate (BA) classifier is insufficient to classify a packet, when peering routers do not have CoS bits marked, or the peering router's marking is untrusted.

- Requirements on page 103
- Overview on page 103
- Configuration on page 104
- Verification on page 107

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Overview

A classifier is a software operation that inspects a packet as it enters the router or switch. The packet header contents are examined, and this examination determines how the packet is treated when the network becomes too busy to handle all of the packets and you want your devices to drop packets intelligently, instead of dropping packets indiscriminately. One common way to detect packets of interest is by source port number. The TCP port numbers 80 and 12345 are used in this example, but many other matching criteria for packet detection are available to multifield classifiers, using firewall filter match conditions. The configuration in this example specifies that TCP packets with source port 80 are classified into the BE-data forwarding class and queue number 0. TCP packets with source port 12345 are classified into the Premium-data forwarding class and queue number 1.

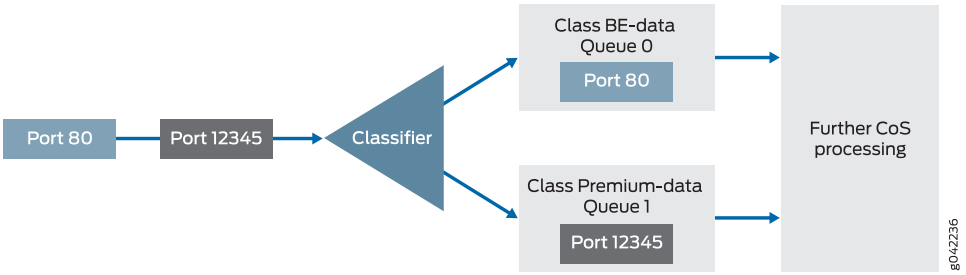
Multifield classifiers are typically used at the network edge as packets enter an autonomous system (AS).

In this example, you configure the firewall filter mf-classifier and specify some custom forwarding classes on Device R1. In specifying the custom forwarding classes, you also associate each class with a queue.

The classifier operation is shown in Figure 14 on page 104.

Copyright © 2019, Juniper Networks, Inc.

Figure 14: Multifield Classifier Based on TCP Source Ports

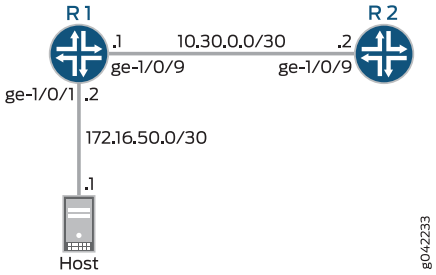


You apply the multifield classifier's firewall filter as an input filter on each customer-facing or host-facing interface that needs the filter. The incoming interface is ge-1/0/1 on Device R1. The classification and queue assignment is verified on the outgoing interface. The outgoing interface is Device R1's ge-1/0/9 interface.

Topology

Figure 15 on page 104 shows the sample network.

Figure 15: Multifield Classifier Scenario



"CLI Quick Configuration" on page 104 shows the configuration for all of the Juniper Networks devices in Figure 15 on page 104.

The section "Step-by-Step Procedure" on page 105 describes the steps on Device R1.

Classifiers are described in more detail in the following Juniper Networks Learning Byte video.

Video: Class of Service Basics, Part 2: Classification Learning Byte

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from the configuration mode.

Device R1

```
set interfaces ge-1/0/1 description to-host
```


Copyright © 2019, Juniper Networks, Inc.

```
set interfaces ge-1/0/1 unit 0 family inet filter input mf-classifier
set interfaces ge-1/0/1 unit 0 family inet address 172.16.50.2/30
set interfaces ge-1/0/9 description to-R2
set interfaces ge-1/0/9 unit 0 family inet address 10.30.0.1/30
set class-of-service forwarding-classes class BE-data queue-num 0
set class-of-service forwarding-classes class Premium-data queue-num 1
set class-of-service forwarding-classes class Voice queue-num 2
set class-of-service forwarding-classes class NC queue-num 3
set firewall family inet filter mf-classifier term BE-data from protocol tcp
set firewall family inet filter mf-classifier term BE-data from port 80
set firewall family inet filter mf-classifier term BE-data then forwarding-class BE-data
set firewall family inet filter mf-classifier term Premium-data from protocol tcp
set firewall family inet filter mf-classifier term Premium-data from port 12345
set firewall family inet filter mf-classifier term Premium-data then forwarding-class
    Premium-data
set firewall family inet filter mf-classifier term accept-all-else then accept
```

Device R2

```
set interfaces ge-1/0/9 description to-R1
set interfaces ge-1/0/9 unit 0 family inet address 10.30.0.2/30
```

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Device R1:

1. Configure the device interfaces.

```
[edit interfaces]
user@R1# set ge-1/0/1 description to-host
user@R1# set ge-1/0/1 unit 0 family inet address 172.16.50.2/30
user@R1# set ge-1/0/9 description to-R2
user@R1# set ge-1/0/9 unit 0 family inet address 10.30.0.1/30
```

2. Configure the custom forwarding classes and associated queue numbers.

```
[edit class-of-service forwarding-classes]
user@R1# set BE-data queue-num 0
user@R1# set Premium-data queue-num 1
user@R1# set Voice queue-num 2
user@R1# set NC queue-num 3
```

3. Configure the firewall filter term that places TCP traffic with a source port of 80 (HTTP traffic) into the BE-data forwarding class, associated with queue 0.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term BE-data from protocol tcp
user@R1# set term BE-data from port 80
user@R1# set term BE-data then forwarding-class BE-data
```

Copyright © 2019, Juniper Networks, Inc.

4. Configure the firewall filter term that places TCP traffic with a source port of 12345 into the Premium-data forwarding class, associated with queue 1.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term Premium-data from protocol tcp
user@R1# set term Premium-data from port 12345
user@R1# set term Premium-data then forwarding-class Premium-data
```

5. At the end of your firewall filter, configure a default term that accepts all other traffic.

   Otherwise, all traffic that arrives on the interface and is not explicitly accepted by the firewall filter is discarded.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term accept-all-else then accept
```

6. Apply the firewall filter to the ge-1/0/1 interface as an input filter.

```
[edit interfaces]
user@R1# set ge-1/0/1 unit 0 family inet filter input mf-classifier
```

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

```
user@R1# show interfaces
ge-1/0/1 {
   description to-host;
   unit 0 {
      family inet {
         filter {
            input mf-classifier;
         }
         address 172.16.50.2/30;
      }
   }
}
ge-1/0/9 {
   description to-R2;
   unit 0 {
      family inet {
         address 10.30.0.1/30;
      }
   }
}
```

```
user@R1# show class-of-service
forwarding-classes {
```


Copyright © 2019, Juniper Networks, Inc.

```
    class BE-data queue-num 0;
    class Premium-data queue-num 1;
    class Voice queue-num 2;
    class NC queue-num 3;
}
```

```
user@R1# show firewall
family inet {
    filter mf-classifier {
        term BE-data {
            from {
                protocol tcp;
                port 80;
            }
            then forwarding-class BE-data;
        }
        term Premium-data {
            from {
                protocol tcp;
                port 12345;
            }
            then forwarding-class Premium-data;
        }
        term accept-all-else {
            then accept;
        }
    }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Checking the CoS Settings on page 107
- Sending TCP Traffic into the Network and Monitoring the Queue Placement on page 108

Checking the CoS Settings

Purpose Confirm that the forwarding classes are configured correctly.

Action From Device R1, run the **show class-of-service forwardng-classes** command.

```
user@R1> show class-of-service forwarding-class

Forwarding class                          ID      Queue  Restricted queue  Fabric
priority  Policing priority   SPU priority
  BE-data                                  0      0          0             low
            normal             low
  Premium-data                             1      1          1             low
            normal             low
```

```
Voice                                   2        2          2              low
          normal           low
NC                                      3        3          3              low
          normal           low
```

Meaning The output shows the configured custom classifier settings.

Sending TCP Traffic into the Network and Monitoring the Queue Placement

Purpose Make sure that the traffic of interest is sent out the expected queue.

Action

1. Clear the interface statistics on Device R1's outgoing interface.

   ```
   user@R1> clear interfaces statistics ge-1/0/9
   ```

2. Use a traffic generator to send 50 TCP port 80 packets to Device R2 or to some other downstream device.

3. On Device R1, check the queue counters.

   Notice that you check the queue counters on the downstream output interface, not on the incoming interface.

   ```
   user@R1> show interfaces extensive ge-1/0/9 | find "Queue counters"

   Queue counters:      Queued packets  Transmitted packets     Dropped packets
     0                                 50                    50
   0
     1                                  0                    57
   0
     2                                  0                     0
   0
     3                                  0                     0
   0
   ```

4. Use a traffic generator to send 50 TCP port 12345 packets to Device R2 or to some other downstream device.

   ```
   [root@host]# hping 172.16.60.1 -c 50 -s 12345 -k
   ```

5. On Device R1, check the queue counters.

   ```
   user@R1> show interfaces extensive ge-1/0/9 | find "Queue counters"

   Queue counters:      Queued packets  Transmitted packets     Dropped packets
     0                                 50                    50
   0
     1                                 50                    57
   0
   ```


Copyright © 2019, Juniper Networks, Inc.

```
   2                                  0                  0
0
   3                                  0                  0
0
```

Meaning The output shows that the packets are classified correctly. When port 80 is used in the TCP packets, queue 0 is incremented. When port 12345 is used, queue 1 is incremented.

Related Documentation

- *Example: Configuring a Two-Rate Three-Color Policer*

Example: Classifying Packets Based on Their Destination Address

This example shows how to classify packets based on their destination address by using a multifield classifier.

- Requirements on page 109
- Overview on page 109
- Configuration on page 109

Requirements

No special configuration beyond device initialization is required before configuring this example.

Overview

In this example you configure a multifield classifier (firewall filter) that ensures that all IPv4 packets destined for the **10.10.10.0/24** network are placed into the **platinum** forwarding class. This assignment occurs regardless of the received CoS bit values in the packet.

You then apply this filter to the inbound interface **so-1/2/2.0** and verify your configuration is attached to the correct interface, issue the **show interfaces filters** command..

Configuration

CLI Quick Configuration To quickly configure the multifield classifer (firewall filter), copy the following commands to a text file, remove any line breaks, and then paste the commands into the CLI.

```
set firewall family inet filter set-FC-to-platinum term match-a-single-route from
   destination-address 10.10.10.0/24
set firewall family inet filter set-FC-to-platinum term match-a-single-route then
   forwarding-class platinum
set firewall family inet filter set-FC-to-platinum term match-a-single-route then accept
set firewall family inet filter set-FC-to-platinum term accept-all then accept
set interfaces so-1/2/2 unit 0 family inet filter input set-FC-to-platinum
```

Copyright © 2019, Juniper Networks, Inc.

Configuring Firewall Filter

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see Using the CLI Editor in Configuration Mode. To configure the multifield classifier (firewall filter):

1. Create and configure the multifield classifier (firewall filter).

```
[edit firewall family inet filter set-FC-to-platinum]
set term match-a-single-route from destination-address 10.10.10.0/24
set term match-a-single-route then forwarding-class platinum
set term match-a-single-route then accept
set term accept-all then accept
```

2. Apply the classifier to the interface.

```
[edit interfaces]
set interfaces so-1/2/2 unit 0 family inet filter input set-FC-to-platinum
```

Results

Confirm your configuration by entering the **show firewall** and **show interfaces** commands from configuration mode. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show firewall
```

```
filter set-FC-to-platinum {
        term match-a-single-route {
            from {
                destination-address {
                    10.10.10.0/24;
                }
            }
            then {
                forwarding-class platinum;
                accept;
            }
        }
```

```
user@host# show interfaces
```

```
so-1/2/2 {
    unit 0 {
        family inet {
            filter {
                input set-FC-to-platinum;
            }
```


Copyright © 2019, Juniper Networks, Inc.

```
        }
    }
}
```

If you are done configuring the device, enter commit from configuration mode.

Related Documentation

- Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 97
- Configuring Multifield Classifiers on page 98

Example: Configuring and Verifying a Complex Multifield Filter

In this example, SIP signaling (VoIP) messages use TCP/UDP, port 5060, and RTP media channels use UDP with port assignments from 16,384 through 32,767. See the following sections:

- Configuring a Complex Multifield Filter on page 111
- Verifying a Complex Multifield Filter on page 113

Configuring a Complex Multifield Filter

To configure the multifield filter, perform the following actions:

- Classify SIP signaling messages (VoIP network control traffic) as NC with a firewall filter.
- Classify VoIP traffic as EF with the same firewall filter.
- Police all remaining traffic with IP precedence **0** and make it BE.
- Police BE traffic to 1 Mbps with excess data marked with PLP high.
- Apply the firewall filter with policer to the interface.

The firewall filter called **classify** matches on the transport protocol and ports identified in the incoming packets and classifies packets into the forwarding classes specified by your criteria.

The first term, **sip**, classifies SIP signaling messages as network control messages. The **port** statement matches any source port or destination port (or both) that is coded to 5060.

Classifying SIP Signaling Messages

```
firewall {
   family inet {
      filter classify {
         interface-specific;
         term sip {
            from {
```

Copyright © 2019, Juniper Networks, Inc.

```
            protocol [ udp tcp ];
            port 5060;
          }
          then {
            forwarding-class network-control;
            accept;
          }
        }
      }
    }
  }
```

The second term, **rtp**, classifies VoIP media channels that use UDP-based transport.

Classifying VoIP Channels That Use UDP

```
term rtp {
  from {
    protocol udp;
    port 16384-32767;
  }
  then {
    forwarding-class expedited-forwarding;
    accept;
  }
}
```

The policer's burst tolerance is set to the recommended value for a low-speed interface, which is ten times the interface MTU. For a high-speed interface, the recommended burst size is the transmit rate of the interface times 3 to 5 milliseconds.

Configuring the Policer

```
policer be-policer {
  if-exceeding {
    bandwidth-limit 1m;
    burst-size-limit 15k;
  }
  then loss-priority high;
}
```

The third term, **be**, ensures that all remaining traffic is policed according to a bandwidth restriction.

Policing All Remaining Traffic

```
term be {
  then policer be-policer;
}
```

The **be** term does not include a **forwarding-class** action modifier. Furthermore, there is no explicit treatment of network control (NC) traffic provided in the **classify** filter. You can configure explicit classification of NC traffic and all remaining IP traffic, but you do

Copyright © 2019, Juniper Networks, Inc.

not need to, because the default IP precedence classifier correctly classifies the remaining traffic.

Apply the **classify** classifier to the **fe-0/0/2** interface:

Applying the Classifier

```
interfaces {
  fe-0/0/2 {
    unit 0 {
      family inet {
        filter {
          input classify;
        }
        address 10.12.0.13/30;
      }
    }
  }
}
```

Verifying a Complex Multifield Filter

Before the configuration is committed, display the default classifiers in effect on the interface using the **show class-of-service interface *interface-name*** command. The display confirms that the **ipprec-compatibility** classifier is in effect by default.

Verifying Default Classification

```
user@host> show class-of-service fe-0/0/2
```

```
Physical interface: fe-0/0/2, Index: 135
Queues supported: 8, Queues in use: 4
  Scheduler map: <default>, Index: 2032638653

  Logical interface: fe-0/0/2.0, Index: 68
    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
```

To view the default classifier mappings, use the **show class-of-service classifier name *name*** command. The highlighted output confirms that traffic with IP precedence setting of 0 is correctly classified as BE, and NC traffic, with precedence values of 6 or 7, is properly classified as NC.

Displaying Default Classifier Mappings

```
user@host> show class-of-service classifier name ipprec-compatibility
```

```
Classifier: ipprec-compatibility, Code point type: inet-precedence, Index: 12
  Code point          Forwarding class                    Loss priority
  000                 best-effort                         low
  001                 best-effort                         high
  010                 best-effort                         low
```

Copyright © 2019, Juniper Networks, Inc.

```
011                 best-effort                     high
100                 best-effort                     low
101                 best-effort                     high
110                 network-control                 low
111                 network-control                 high
```

After your configuration is committed, verify that your multifield classifier is working correctly. You can monitor the queue counters for the router device's ***egress*** interface used when forwarding traffic received from the peer. Displaying the queue counters for the ingress interface (**fe-0/0/2**) does not allow you to check your ingress classification, because queuing generally occurs only at egress in the Junos OS. (Ingress queuing is supported on Gigabit Ethernet IQ2 PICs and Enhanced IQ2 PICs only.)

To verify the operation of the multifield filter:

1. To determine which egress interface is used for the traffic, use the **traceroute** command.
2. After you identify the egress interface, clear its associated queue counters by issuing the **clear interfaces statistics** ***interface-name*** command.
3. Confirm the default forwarding class-to-queue number assignment. This allows you to predict which queues are used by the VoIP, NC, and other traffic. To do this, issue the **show class-of-service forwarding-class** command.
4. Display the queue counts on the interface by issuing the **show interfaces queue** command.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 4

Controlling Network Access with Traffic Policing

- Controlling Network Access Using Traffic Policing Overview on page 115
- Effect of Two-Color Policers on Shaping Rate Changes on page 120
- Configuring Policers Based on Logical Interface Bandwidth on page 122
- Example: Limiting Inbound Traffic at Your Network Border by Configuring an Ingress Single-Rate Two-Color Policer on page 124
- Example: Performing CoS at an Egress Network Boundary by Configuring an Egress Single-Rate Two-Color Policer on page 133
- Example: Limiting Inbound Traffic Within Your Network by Configuring an Ingress Single-Rate Two-Color Policer and Configuring Multifield Classifiers on page 142
- Example: Limiting Outbound Traffic Within Your Network by Configuring an Egress Single-Rate Two-Color Policer and Configuring Multifield Classifiers on page 155
- Overview of Tricolor Marking Architecture on page 171
- Enabling Tricolor Marking and Limitations of Three-Color Policers on page 173
- Configuring and Applying Tricolor Marking Policers on page 175
- Configuring Single-Rate Tricolor Marking on page 180
- Configuring Two-Rate Tricolor Marking on page 184
- Example: Configuring and Verifying Two-Rate Tricolor Marking on page 187
- Applying Firewall Filter Tricolor Marking Policers to Interfaces on page 194
- Policer Overhead to Account for Rate Shaping in the Traffic Manager on page 195

Controlling Network Access Using Traffic Policing Overview

- Congestion Management for IP Traffic Flows on page 116
- Traffic Limits on page 116
- Traffic Color Marking on page 117
- Forwarding Classes and PLP Levels on page 119
- Policer Application to Traffic on page 119

Copyright © 2019, Juniper Networks, Inc.

Congestion Management for IP Traffic Flows

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

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

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

You can apply a policer to inbound or outbound interface traffic. Policers applied to inbound traffic help to conserve resources by dropping traffic that does not need to be routed through a network. Dropping inbound traffic also helps to thwart denial-of-service (DoS) attacks. Policers applied to outbound traffic control the bandwidth used.

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

Traffic Limits

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

In the token-bucket model, the bucket represents the rate-limiting function of the policer. Tokens are added to the bucket at a fixed rate, but once the specified depth of the bucket is reached, tokens allocated after cannot be stored and used. Each token represents a “credit” for some number of bits, and tokens in the bucket are “cashed in” for the ability to transmit or receive traffic at the interface. When sufficient tokens are present in the bucket, a traffic flow continues unrestricted. Otherwise, packets might be dropped or else re-marked with a lower forwarding class, a higher packet loss priority (PLP) level, or both.

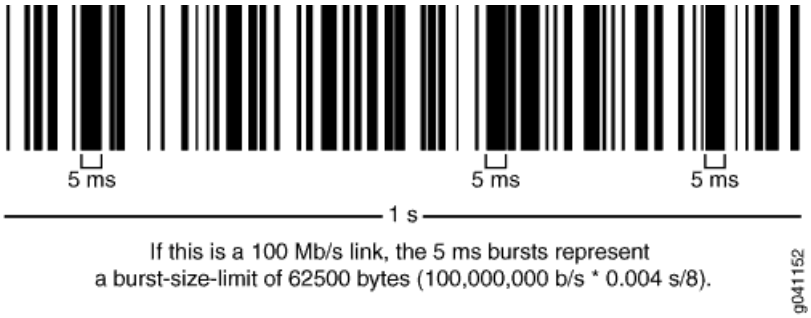
- The rate at which tokens are added to the bucket represents the highest average transmit or receive rate in bits per second allowed for a given service level. You specify this highest average traffic rate as the *bandwidth limit* of the policer. If the traffic arrival

Copyright © 2019, Juniper Networks, Inc.

rate (or fixed bits-per-second) is so high that at some point insufficient tokens are present in the bucket, then the traffic flow is no longer conforming to the traffic limit. During periods of relatively low traffic (traffic that arrives at or departs from the interface at average rates below the token arrival rate), unused tokens accumulate in the bucket.

- The depth of the bucket in bytes controls the amount of back-to-back bursting allowed. You specify this factor as the *burst-size limit* of the policer. This second limit affects the average transmit or receive rate by limiting the number of bytes permitted in a transmission burst for a given interval of time. Bursts exceeding the current burst-size limit are dropped until there are sufficient tokens available to permit the burst to proceed.

Figure 16: Network Traffic and Burst Rates



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

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

Traffic Color Marking

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

- *Single-rate two-color*—A two-color marking policer (or "policer" when used without qualification) meters the traffic stream and classifies packets into two categories of packet loss priority (PLP) according to a configured bandwidth and burst-size limit. You can mark packets that exceed the bandwidth and burst-size limit in some way, or simply discard them.

  A policer is most useful for metering traffic at the port (physical interface) level.

- *Single-rate three-color*—This type of policer is defined in RFC 2697, *A Single Rate Three Color Marker*, as part of an assured forwarding (AF) per-hop-behavior (PHB) classification system for a Differentiated Services (DiffServ) environment. This type of policer meters traffic based on the configured committed information rate (CIR),

Copyright © 2019, Juniper Networks, Inc.

committed burst size (CBS), and the excess burst size (EBS). Traffic is marked as belonging to one of three categories (green, yellow, or red) based on whether the packets arriving are below the CBS (green), exceed the CBS (yellow) but not the EBS, or exceed the EBS (red).

A single-rate three-color policer is most useful when a service is structured according to packet length and not peak arrival rate.

- *Two-rate three-color*—This type of policer is defined in RFC 2698, *A Two Rate Three Color Marker*, as part of an assured forwarding (AF) per-hop-behavior (PHB) classification system for a Differentiated Services (DiffServ) environment. This type of policer meters traffic based on the configured CIR and peak information rate (PIR), along with their associated burst sizes, the CBS and *peak burst size* (PBS). Traffic is marked as belonging to one of three categories (green, yellow, or red) based on whether the packets arriving are below the CIR (green), exceed the CIR (yellow) but not the PIR, or exceed the PIR (red).

  A two-rate three-color policer is most useful when a service is structured according to arrival rates and not necessarily packet length.

Policer actions are implicit or explicit and vary by policer type. The term *Implicit* means that Junos assigns the loss-priority automatically. Table 16 on page 118 describes the policer actions.

Table 16: Policer Actions

Policer	Marking	Implicit Action	Configurable Action
Single-rate two-color	Green (Conforming)	Assign low loss priority	None
	Red (Nonconforming)	None	Assign low or high loss priority, assign a forwarding class, or discard On some platforms, you can assign medium-low or medium-high loss priority
Single-rate three-color	Green (Conforming)	Assign low loss priority	None
	Yellow (Above the CIR and CBS)	Assign medium-high loss priority	None
	Red (Above the EBS)	Assign high loss priority	Discard

Copyright © 2019, Juniper Networks, Inc.

Table 16: Policer Actions (continued)

Policer	Marking	Implicit Action	Configurable Action
Two-rate three-color	Green (Conforming)	Assign low loss priority	None
	Yellow (Above the CIR and CBS)	Assign medium-high loss priority	None
	Red (Above the PIR and PBS)	Assign high loss priority	Discard

Forwarding Classes and PLP Levels

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

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

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

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

Policer Application to Traffic

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

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

- You can configure a standard stateless firewall filter that specifies the **policer *policer-name*** nonterminating action or the **three-color-policer (single-rate | two-rate) *policer-name*** nonterminating action. When you apply the standard filter to

Copyright © 2019, Juniper Networks, Inc.

the input or output at a logical interface, the policer is applied to all packets of the filter-specific protocol family that match the conditions specified in the filter configuration.

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

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

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

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

Related Documentation

- *Stateless Firewall Filter Overview*.
- *Traffic Policer Types*
- *Order of Policer and Firewall Filter Operations*
- Packet Flow Through the Junos OS CoS Process Overview on page 17

Effect of Two-Color Policers on Shaping Rate Changes

When you configure a change in shaping rate, it is important to consider the effect on the bandwidth limit. Whenever the shaping rate changes, the bandwidth limit is adjusted based on whether a logical interface (unit) or bandwidth percentage policer is configured.

When a logical interface bandwidth policer is configured, the order of priority for the shaping rate (if configured at that level) is:

- The shaping rate applied to the logical interface (unit).
- The shaping rate applied to the physical interface (port).
- The physical interface speed.

When a bandwidth percentage policer is configured, the order of priority for the shaping rate (if configured at that level) is:

- The shaping rate applied to the physical interface (port).
- The physical interface speed.

These guidelines must be kept in mind when calculating the logical link speed and link speed from the configured shaping rate, which determines the rate-limited bandwidth after the policer is applied.

Copyright © 2019, Juniper Networks, Inc.

In the following configuration, for example, a shaping rate has been configured for the logical interface, but a bandwidth percentage policer is also configured and applied to the same logical interface. Therefore policing is based on the physical interface speed of 1 Gbps.

```
[edit interfaces]
ge-0/1/0 {
   per-unit-scheduler;
   vlan-tagging;
   unit 0 {
      vlan-id 1;
      family inet {
         policer {
            output policer_test;
         }
         address 10.0.7.1/24;
      }
   }
}

[edit firewall]
policer policer_test {
   if-exceeding {
      bandwidth-percent 75;
      burst-size-limit 256k;
   }
   then discard;
}

[edit]
class-of-service {
   interfaces {
      ge-0/1/0 {
         unit 0 {
            shaping-rate 15m;
         }
      }
   }
}
```

Related Documentation

- Configuring Policers Based on Logical Interface Bandwidth on page 122

Copyright © 2019, Juniper Networks, Inc.

Configuring Policers Based on Logical Interface Bandwidth

When you configure a policer as a percentage (using the **bandwidth-percent** statement), the bandwidth is calculated as a percentage of either the physical interface media rate or the logical interface shaping rate.

- To specify that the bandwidth be calculated based on the logical interface shaping rate and not the physical interface media rate, set the **logical-bandwidth-policer** option at the **[edit firewall]** hierarchy level. Next,, specify the **shaping-rate** for the logical interfaces under the **[edit class-of-service]** hierarchy level and apply the policer to the logical interfaces..
- If a shaping rate is not configured for the logical interface, the physical interface media rate is used, even if you include the **logical-bandwidth-policer**. You can configure the shaping rate on the logical interface using class-of-service statements.

The following example configures and applies a logical bandwidth policer rate to two logical interfaces on interface **ge-0/2/7**. The policed rate on **unit 0** is 2 Mbps (50 percent of 4 Mbps) and the policed rate on **unit 1** is 1 Mbps (50 percent of 2 Mbps).

To configure and apply this policer:

1. Create and configure the policer.
   a. Create the policer.

      ```
      [edit]
      user@host# edit firewall policer Logical_Policer
      ```

   b. Specify that the policer is based on the shaping rate of the logical interface.

      ```
      [edit firewall policer Logical_Policer]
      user@host# set logical-bandwidth-policer
      ```

   c. Configure the rate limits for the policer.

      ```
      [edit firewall policer Logical_Policer]
      user@host# set if-exceeding bandwidth-limit 50
      user@host# set burst-size-limit 125k
      ```

   d. Configure the policer to discard packets that exceed the specified rate limits.

      ```
      [edit firewall policer Logical_Policer]
      user@host# set then discard
      ```

2. Specify the shaping-rate for each logical interface.

   ```
   {edit}
   user@host# edit class-of-service interfaces ge-0/2/7
   user@host# set unit 0 shaping-rate 4m
   user@host# set unit 1 shaping-rate 2m
   ```


Copyright © 2019, Juniper Networks, Inc.

3. Apply the policer to the logical interfaces.

   - Enable scheduling on logical interfaces.

     ```
     [edit]
     user@host# edit interfaces ge-0/2/7
     user@host# set per-unit-scheduler
     ```

   - Enable the reception and transmission of 802.1Q VLAN-tagged frames on the interface.

     ```
     [edit interfaces ge-0/2/7]
     user@host# set vlan-tagging
     ```

   - Apply the policer to the first logical interface.

     ```
     [edit interfaces ge-0/2/7]
     user@host# set unit 0 vlan-id 100 family inet policer input Logical_Policer
     user@host# set unit 0 vlan-id 100 family inet policer output Logical_Policer
     user@host# set unit 0 vlan-id 100 family inet address 172.16.1.1/30
     ```

   - Apply the policer to the second logical interface.

     ```
     [edit interfaces ge-0/2/7]
     user@host# set unit 1 vlan-id 200 family inet policer input Logical_Policer
     user@host# set unit 1 vlan-id 200 family inet policer output Logical_Policer
     user@host# set unit 1 vlan-id 200 family inet address 172.26.1.1/30
     ```

4. Confirm your configuration.

   ```
   [edit]
   user@host# show firewall
   ```

   ```
   policer Logical_Policer {
       logical-bandwidth-policer;
       if-exceeding {
           bandwidth-percent 50;
           burst-size-limit 125k;
       }
       then discard;
   }
   ```

   ```
   [edit]
   user@host# show class-of-service interfaces ge-0/2/7
   ```

   ```
   unit 0 {
       shaping-rate 4m;
   }
   unit 1 {
       shaping-rate 2m;
   }
   ```

Copyright © 2019, Juniper Networks, Inc.

```
[edit]
user@host# show interfaces ge-0/2/7
```

```
per-unit-scheduler;
vlan-tagging;
unit 0 {
    vlan-id 100;
    family inet {
        policer {
            input Logical_Policer;
            output Logical_Policer;
        }
        address 172.16.1.1/30;
    }
}
unit 1 {
    vlan-id 200;
    family inet {
        policer {
            input Logical_Policer;
            output Logical_Policer;
        }
        address 172.26.1.1/30;
    }
}
```

5. Save the configuration.

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

Related Documentation

- Controlling Network Access Using Traffic Policing Overview on page 115
- logical-bandwidth-policer on page 1190
- shaping-rate (Applying to an Interface) on page 1265

Example: Limiting Inbound Traffic at Your Network Border by Configuring an Ingress Single-Rate Two-Color Policer

This example shows you how to configure an ingress single-rate two-color policer to filter incoming traffic. The policer enforces the class-of-service (CoS) strategy for in-contract and out-of-contract traffic. You can apply a single-rate two-color policer to incoming packets, outgoing packets, or both. This example applies the policer as an input (ingress) policer. The goal of this topic is to provide you with an introduction to policing by using a example that shows traffic policing in action.

Policers use a concept known as a token bucket to allocate system resources based on the parameters defined for the policer. A thorough explanation of the token bucket concept and its underlying algorithms is beyond the scope of this document. For more information about traffic policing, and CoS in general, refer to *QOS-Enabled*

Copyright © 2019, Juniper Networks, Inc.

Networks—Tools and Foundations by Miguel Barreiros and Peter Lundqvist. This book is available at many online booksellers and at www.juniper.net/books.

- Requirements on page 125
- Overview on page 125
- Configuration on page 127
- Verification on page 132

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Overview

Single-rate two-color policing enforces a configured rate of traffic flow for a particular service level by applying implicit or configured actions to traffic that does not conform to the limits. When you apply a single-rate two-color policer to the input or output traffic at an interface, the policer meters the traffic flow to the rate limit defined by the following components:

- Bandwidth limit—The average number of bits per second permitted for packets received or transmitted at the interface. You can specify the bandwidth limit as an absolute number of bits per second or as a percentage value from 1 through 100. If a percentage value is specified, the effective bandwidth limit is calculated as a percentage of either the physical interface media rate or the logical interface configured shaping rate.
- Burst-size limit—The maximum size permitted for bursts of data. Burst sizes are measured in bytes. We recommend two formulas for calculating burst size:

  Burst size = bandwidth x allowable time for burst traffic / 8

  Or

  Burst size = interface mtu x 10

  For information about configuring the burst size, see *Determining Proper Burst Size for Traffic Policers*.

NOTE: **There is a finite buffer space for an interface. In general, the estimated total buffer depth for an interface is about 125 ms.**

For a traffic flow that conforms to the configured limits (categorized as green traffic), packets are implicitly marked with a packet loss priority (PLP) level of low and are allowed to pass through the interface unrestricted.

Copyright © 2019, Juniper Networks, Inc.

For a traffic flow that exceeds the configured limits (categorized as red traffic), packets are handled according to the traffic-policing actions configured for the policer. This example discards packets that burst over the 15 KBps limit.

To rate-limit Layer 3 traffic, you can apply a two-color policer in the following ways:

- Directly to a logical interface, at a specific protocol level.
- As the action of a standard stateless firewall filter that is applied to a logical interface, at a specific protocol level. This is the technique used in this example.

To rate-limit Layer 2 traffic, you can apply a two-color policer as a logical interface policer only. You cannot apply a two-color policer to Layer 2 traffic through a firewall filter.

CAUTION: You can choose either bandwidth-limit or bandwidth percent within the policer, as they are mutually exclusive. You cannot configure a policer to use bandwidth percent for aggregate, tunnel, and software interfaces.

In this example, the host is a traffic generator emulating a webserver. Devices R1 and R2 are owned by a service provider. The webserver is accessed by users on Device Host2. Device Host1 will be sending traffic with a source TCP HTTP port of 80 to the users. A single-rate two-color policer is configured and applied to the interface on Device R1 that connects to Device Host1. The policer enforces the contractual bandwidth availability made between the owner of the webserver and the service provider that owns Device R1 for the web traffic that flows over the link that connects Device Host1 to Device R1.

In accordance with the contractual bandwidth availability made between the owner of the webserver and the service provider that owns Devices R1 and R2, the policer will limit the HTTP port 80 traffic originating from Device Host1 to using 700 Mbps (70 percent) of the available bandwidth with an allowable burst rate of 10 x the MTU size of the gigabit Ethernet interface between the host Device Host1 and Device R1.

NOTE: In a real-world scenario you would probably also rate limit traffic for a variety of other ports such as FTP, SFTP, SSH, TELNET, SMTP, IMAP, and POP3 because they are often included as additional services with web hosting services.

NOTE: You need to leave some additional bandwidth available that is not rate limited for network control protocols such as routing protocols, DNS, and any other protocols required to keep network connectivity operational. This is why the firewall filter has a final accept condition on it.

Topology

This example uses the topology in Figure 17 on page 127.

Copyright © 2019, Juniper Networks, Inc.

Figure 17: Single-Rate Two-Color Policer Scenario

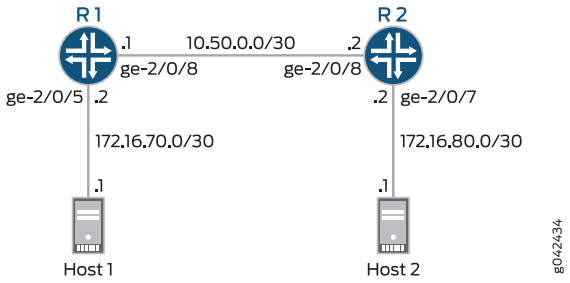
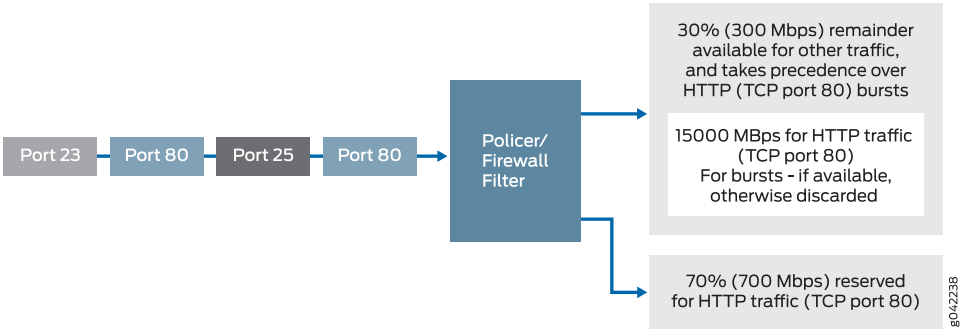


Figure 18 on page 127 shows the policing behavior.

Figure 18: Traffic Limiting in a Single-Rate Two-Color Policer Scenario



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.

Device R1

```
set interfaces ge-2/0/5 description to-Host
set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
set interfaces ge-2/0/5 unit 0 family inet filter input mf-classifier
set interfaces ge-2/0/8 description to-R2
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.13.1/32
set firewall policer discard if-exceeding bandwidth-limit 700m
set firewall policer discard if-exceeding burst-size-limit 15k
set firewall policer discard then discard
set firewall family inet filter mf-classifier term t1 from protocol tcp
set firewall family inet filter mf-classifier term t1 from port 80
set firewall family inet filter mf-classifier term t1 then policer discard
set firewall family inet filter mf-classifier term t2 then accept
set protocols ospf area 0.0.0.0 interface ge-2/0/5.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
```

Copyright © 2019, Juniper Networks, Inc.

```
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Device R2

```
set interfaces ge-2/0/8 description to-R1
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set interfaces ge-2/0/7 description to-Host
set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.14.1/32
set protocols ospf area 0.0.0.0 interface ge-2/0/7.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Device R1:

1. Configure the device interfaces.

```
[edit interfaces]
user@R1# set ge-2/0/5 description to-Host
user@R1# set ge-2/0/5 unit 0 family inet address 172.16.70.2/30
user@R1# set ge-2/0/8 description to-R2
user@R1# set ge-2/0/8 unit 0 family inet address 10.50.0.1/30
user@R1# set lo0 unit 0 description looback-interface
user@R1# set lo0 unit 0 family inet address 192.168.13.1/32
```

2. Apply the firewall filter to interface ge-2/0/5 as an input filter.

```
[edit interfaces ge-2/0/5 unit 0 family inet]
user@R1# set filter input mf-classifier
```

3. Configure the policer to rate-limit to a bandwidth of 700 Mbps and a burst size of 15000 KBps for HTTP traffic (TCP port 80).

```
[edit firewall policer discard]
user@R1# set if-exceeding bandwidth-limit 700m
user@R1# set if-exceeding burst-size-limit 15k
```

4. Configure the policer to discard packets in the red traffic flow.

```
[edit firewall policer discard]
user@R1# set then discard
```

5. Configure the two conditions of the firewall to accept all TCP traffic to port HTTP (port 80).

Copyright © 2019, Juniper Networks, Inc.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term t1 from protocol tcp
user@R1# set term t1 from port 80
```

6. Configure the firewall action to rate-limit HTTP TCP traffic using the policer.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term t1 then policer discard
```

7. At the end of the firewall filter, configure a default action that accepts all other traffic.

   Otherwise, all traffic that arrives on the interface and is not explicitly accepted by the firewall is discarded.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term t2 then accept
```

8. Configure OSPF.

```
[edit protocols ospf]
user@R1# set area 0.0.0.0 interface ge-2/0/5.0 passive
user@R1# set area 0.0.0.0 interface lo0.0 passive
user@R1# set area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

To configure Device R2:

1. Configure the device interfaces.

```
[edit interfaces]
user@R1# set ge-2/0/8 description to-R1
user@R1# set ge-2/0/7 description to-Host
user@R1# set lo0 unit 0 description looback-interface
user@R1# set ge-2/0/8 unit 0 family inet address 10.50.0.2/30
user@R1# set ge-2/0/7 unit 0 family inet address 172.16.80.2/30
user@R1# set lo0 unit 0 family inet address 192.168.14.1/32
```

2. Configure OSPF.

```
[edit protocols ospf]
user@R1# set area 0.0.0.0 interface ge-2/0/7.0 passive
user@R1# set area 0.0.0.0 interface lo0.0 passive
user@R1# set area 0.0.0.0 interface ge-2/0/8.0
```

Copyright © 2019, Juniper Networks, Inc.

Results From configuration mode, confirm your configuration by entering the **show interfaces** , **show firewall**, and **show protocols ospf** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R1# show interfaces
ge-2/0/5 {
   description to-Host;
   unit 0 {
      family inet {
         filter {
            input mf-classifier;
         }
         address 172.16.70.2/30;
      }
   }
}
ge-2/0/8 {
   description to-R2;
   unit 0 {
      family inet {
         address 10.50.0.1/30;
      }
   }
}
lo0 {
   unit 0 {
      description looback-interface;
      family inet {
         address 192.168.13.1/32;
      }
   }
}
```

```
user@R1# show firewall
family inet {
   filter mf-classifier {
      term t1 {
         from {
            protocol tcp;
            port 80;
         }
         then policer discard;
      }
      term t2 {
         then accept;
      }
   }
}
policer discard {
   if-exceeding {
      bandwidth-limit 700m;
      burst-size-limit 15k;
   }
   then discard;
```

 Copyright © 2019, Juniper Networks, Inc.

```
}
```

```
user@R1# show protocols ospf
area 0.0.0.0 {
   interface ge-2/0/5.0 {
      passive;
   }
   interface lo0.0 {
      passive;
   }
   interface ge-2/0/8.0;
}
```

If you are done configuring Device R1, enter **commit** from configuration mode.

```
user@R2# show interfaces
ge-2/0/7 {
   description to-Host;
   unit 0 {
      family inet {
         address 172.16.80.2/30;
      }
   }
}
ge-2/0/8 {
   description to-R1;
   unit 0 {
      family inet {
         address 10.50.0.2/30;
      }
   }
}
lo0 {
   unit 0 {
      description looback-interface;
      family inet {
         address 192.168.14.1/32;
      }
   }
}
```

```
user@R2# show protocols ospf
area 0.0.0.0 {
   interface ge-2/0/7.0 {
      passive;
   }
   interface lo0.0 {
      passive;
   }
   interface ge-2/0/8.0;
}
```

If you are done configuring Device R2, enter **commit** from configuration mode.

Copyright © 2019, Juniper Networks, Inc.

Verification

Confirm that the configuration is working properly.

- Clearing the Counters on page 132
- Sending TCP Traffic into the Network and Monitoring the Discards on page 132

Clearing the Counters

Purpose Confirm that the firewall counters are cleared.

Action On Device R1, run the **clear firewall all** command to reset the firewall counters to 0.

```
user@R1> clear firewall all
```

Sending TCP Traffic into the Network and Monitoring the Discards

Purpose Make sure that the traffic of interest that is sent is rate-limited on the input interface (ge-2/0/5).

Action

1. Use a traffic generator to send 10 TCP packets with a source port of 80.

   The -s flag sets the source port. The -k flag causes the source port to remain steady at 80 instead of incrementing. The -c flag sets the number of packets to 10. The -d flag sets the packet size.

   The destination IP address of 172.16.80.1 belongs to Device Host 2 that is connected to Device R2. The user on Device Host 2 has requested a webpage from Device Host 1 (the webserver emulated by the traffic generator on Device Host 1). The packets that being rate-limited are sent from Device Host 1 in response to the request from Device Host 2.

   NOTE: In this example the policer numbers are reduced to a bandwidth limit of 8 Kbps and a burst size limit of 1500 KBps to ensure that some packets are dropped during this test.

```
[root@host]# hping 172.16.80.1 -c 10 -s 80 -k -d 300
```

```
[User@Host]#  hping 172.16.80.1 -c 10 -s 80 -k -d 350
HPING 172.16.80.1 (eth1 172.16.80.1): NO FLAGS are set, 40 headers + 350 data
 bytes
len=46 ip=172.16.80.1 ttl=62 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=0.5 ms
.
.
.
--- 172.16.80.1 hping statistic ---
```


Copyright © 2019, Juniper Networks, Inc.

```
10 packets transmitted, 6 packets received, 40% packet loss
round-trip min/avg/max = 0.5/3000.8/7001.3 ms
```

2. On Device R1, check the firewall counters by using the **show firewall** command.

```
user@R1> show firewall
```

```
User@R1# run show firewall

Filter: __default_bpdu_filter__

Filter: mf-classifier
Policers:
Name                                                Bytes              Packets
discard-t1                                           1560                4
```

Meaning In Steps 1 and 2 the output from both devices shows that 4 packets were discarded This means that there was at least 8 Kbps of green (in-contract HTTP port 80) traffic and that the 1500 KBps burst option for red out-of-contract HTTP port 80 traffic was exceeded.

Related Documentation

- *Junos OS Routing Protocols and Policies Configuration Guide for Security Devices*

Example: Performing CoS at an Egress Network Boundary by Configuring an Egress Single-Rate Two-Color Policer

This example shows how to configure an egress single-rate two-color policer. Policers use a concept known as a token bucket. The policer enforces the class-of-service (CoS) strategy for in-contract and out-of-contract traffic. You can apply a single-rate two-color policer to incoming packets, outgoing packets, or both. This example applies the policer as an output (egress) policer. This example is an introduction to policing by using an example that shows traffic policing in action.

A thorough explanation of the token bucket concept and its underlying algorithms is beyond the scope of this document. For more information about traffic policing, and CoS in general, refer to *QOS-Enabled Networks—Tools and Foundations* by Miguel Barreiros and Peter Lundqvist. This book is available at many online booksellers and at www.juniper.net/books.

- Requirements on page 134
- Overview on page 134
- Configuration on page 136
- Verification on page 141

Copyright © 2019, Juniper Networks, Inc.

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Overview

Single-rate two-color policing enforces a configured rate of traffic flow for a particular service level by applying implicit or configured actions to traffic that does not conform to the limits. When you apply a single-rate two-color policer to the input or output traffic at an interface, the policer meters the traffic flow to the rate limit defined by the following components:

- Bandwidth limit—The average number of bits per second permitted for packets received or transmitted at the interface. You can specify the bandwidth limit as an absolute number of bits per second or as a percentage value from 1 through 100. If a percentage value is specified, the effective bandwidth limit is calculated as a percentage of either the physical interface media rate or the logical interface configured shaping rate.
- Burst-size limit—The maximum size permitted for bursts of data. Burst sizes are measured in bytes. We recommend two formulas for calculating burst size:

  Burst size = bandwidth x allowable time for burst traffic / 8

  Or

  Burst size = interface mtu x 10

  For information about configuring the burst size, see *Determining Proper Burst Size for Traffic Policers*.

NOTE: **There is a finite buffer space for an interface. In general, the estimated total buffer depth for an interface is about 125 ms.**

For a traffic flow that conforms to the configured limits (categorized as green traffic), packets are implicitly marked with a packet loss priority (PLP) level of low and are allowed to pass through the interface unrestricted.

For a traffic flow that exceeds the configured limits (categorized as red traffic), packets are handled according to the traffic-policing actions configured for the policer. This example discards packets that burst over the 15 KBps limit.

To rate-limit Layer 3 traffic, you can apply a two-color policer in the following ways:

- Directly to a logical interface, at a specific protocol level.
- As the action of a standard stateless firewall filter that is applied to a logical interface, at a specific protocol level. This is the technique used in this example.

Copyright © 2019, Juniper Networks, Inc.

To rate-limit Layer 2 traffic, you can apply a two-color policer as a logical interface policer only. You cannot apply a two-color policer to Layer 2 traffic through a firewall filter.

CAUTION: You can choose either bandwidth-limit or bandwidth percent within the policer, as they are mutually exclusive. You cannot configure a policer to use bandwidth percent for aggregate, tunnel, or software interfaces.

In this example, the host is a traffic generator emulating a webserver. Devices R1 and R2 are owned by a service provider. The webserver is accessed by users behind Device R2. The host will be sending traffic with a source TCP HTTP port of 80 to the users. A single-rate two-color policer is configured and applied to the interface on Device R1 that connects to Device R2. The policer enforces the contractual bandwidth availability made between the owner of the webserver (in this case emulated by the host) and the service provider that owns Devices R1 and R2 for the web traffic that flows over the link that connects Devices R1 and R2.

In accordance with the contractual bandwidth availability made between the owner of the webserver and the service provider that owns Devices R1 and R2, the policer will limit the HTTP port 80 traffic originating from the host to using 700 Mbps (70 percent) of the available bandwidth with an allowable burst rate of 10 x the MTU size of the gigabit Ethernet interface between Devices R1 and R2.

NOTE: In a real-world scenario you would probably also rate-limit traffic for a variety of other ports such as FTP, SFTP, SSH, TELNET, SMTP, IMAP, and POP3 because they are often included as additional services with web hosting services.

NOTE: You need to leave some additional bandwidth available that is not rate-limited for network control protocols such as routing protocols, DNS, and any other protocols required to keep network connectivity operational. This is why the firewall filter has a final accept condition on it.

Topology

This example uses the topology in Figure 17 on page 127.

Copyright © 2019, Juniper Networks, Inc.

Figure 19: Single-Rate Two-Color Policer Scenario

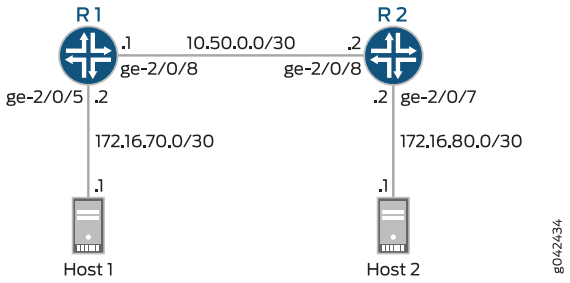
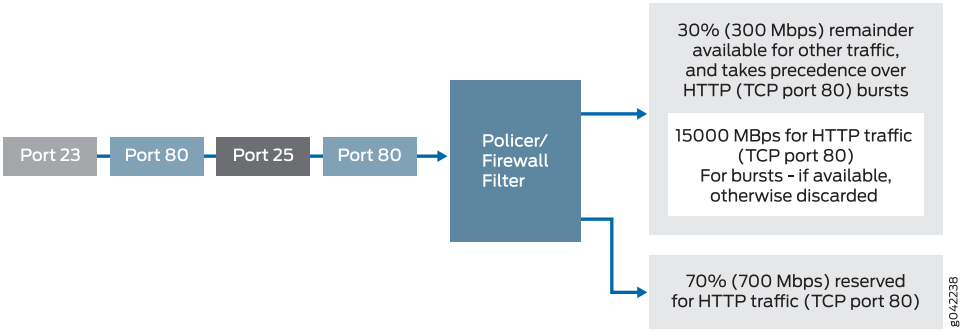


Figure 18 on page 127 shows the policing behavior.

Figure 20: Traffic Limiting in a Single-Rate Two-Color Policer Scenario



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.

Device R1

```
set interfaces ge-2/0/5 description to-Host
set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
set interfaces ge-2/0/8 description to-R2
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
set interfaces ge-2/0/8 unit 0 family inet filter output mf-classifier
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.13.1/32
set firewall policer discard if-exceeding bandwidth-limit 700m
set firewall policer discard if-exceeding burst-size-limit 15k
set firewall policer discard then discard
set firewall family inet filter mf-classifier term t1 from protocol tcp
set firewall family inet filter mf-classifier term t1 from port 80
set firewall family inet filter mf-classifier term t1 then policer discard
set firewall family inet filter mf-classifier term t2 then accept
set protocols ospf area 0.0.0.0 interface ge-2/0/5.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
```


Copyright © 2019, Juniper Networks, Inc.

```
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Device R2

```
set interfaces ge-2/0/7 description to-Host
set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
set interfaces ge-2/0/8 description to-R1
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.14.1/32
set protocols ospf area 0.0.0.0 interface ge-2/0/7.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Device R1:

1. Configure the device interfaces.

```
[edit interfaces]
user@R1#set ge-2/0/5 description to-Host
user@R1#set ge-2/0/5 unit 0 family inet address 172.16.70.2/30
user@R1#set ge-2/0/8 description to-R2
user@R1#set ge-2/0/8 unit 0 family inet address 10.50.0.1/30
user@R1# set lo0 unit 0 description looback-interface
user@R1#set lo0 unit 0 family inet address 192.168.13.1/32
```

2. Configure the policer to rate-limit to a bandwidth of 700 Mbps and a burst size of 15 KBps for HTTP traffic (TCP port 80).

```
[edit firewall policer discard]
user@R1# set if-exceeding bandwidth-limit 700m
user@R1# set if-exceeding burst-size-limit 15k
```

3. Configure the policer to discard packets in the red traffic flow.

```
[edit firewall policer discard]
user@R1# set then discard
```

4. Configure the two conditions of the firewall to accept all TCP traffic to port HTTP (port 80).

```
[edit firewall family inet filter mf-classifier]
user@R1# set term t1 from protocol tcp
user@R1# set term t1 from port 80
```

5. Configure the firewall action to rate-limit HTTP TCP traffic using the policer.

Copyright © 2019, Juniper Networks, Inc.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term t1 then policer discard
```

6. At the end of the firewall filter, configure a default action that accepts all other traffic.

   Otherwise, all traffic that arrives on the interface and is not explicitly accepted by the firewall is discarded.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term t2 then accept
```

7. Apply the firewall filter to interface ge-2/0/8 as an output filter.

```
[edit interfaces ge-2/0/8 unit 0 family inet]
user@R1# set filter output mf-classifier
```

8. Configure OSPF.

```
[edit protocols ospf]
user@R1# set area 0.0.0.0 interface ge-2/0/5.0 passive
user@R1# set area 0.0.0.0 interface lo0.0 passive
user@R1# set area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

To configure Device R2:

1. Configure the device interfaces.

```
[edit interfaces]
set ge-2/0/7 description to-Host
set ge-2/0/7 unit 0 family inet address 172.16.80.2/30
set ge-2/0/8 description to-R1
set ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set lo0 unit 0 description looback-interface
set lo0 unit 0 family inet address 192.168.14.1/32
```

2. Configure OSPF.

```
[edit protocols ospf]
set area 0.0.0.0 interface ge-2/0/7.0 passive
set area 0.0.0.0 interface lo0.0 passive
set area 0.0.0.0 interface ge-2/0/8.0
```

Results

From configuration mode, confirm your configuration by entering the **show interfaces**, **show firewall**, and **show protocols OSPF** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

Copyright © 2019, Juniper Networks, Inc.

```
ge-2/0/5 {
   description to-Host;
   unit 0 {
      family inet {
         address 172.16.70.2/30;
      }
   }
}
ge-2/0/8 {
   description to-R2;
   unit 0 {
      family inet {
         filter {
            output mf-classifier;
         }
         address 10.50.0.1/30;
      }
   }
}
lo0 {
   unit 0 {
      description looback-interface;
      family inet {
         address 192.168.13.1/32;
      }
   }
}
```

```
user@R1# show firewall
family inet {
   filter mf-classifier {
      term t1 {
         from {
            protocol tcp;
            port 80;
         }
         then policer discard;
      }
      term t2 {
         then accept;
      }
   }
}
policer discard {
   if-exceeding {
      bandwidth-limit 700m;
      burst-size-limit 15k;
   }
   then discard;
}
```

```
policer discard {
   if-exceeding {
      bandwidth-limit 700m;
```

Copyright © 2019, Juniper Networks, Inc.

```
      burst-size-limit 15k;
    }
    then discard;
}
```

```
user@R1# show protocols ospf
area 0.0.0.0 {
    interface ge-2/0/5.0 {
        passive;
    }
    interface lo0.0 {
        passive;
    }
    interface ge-2/0/8.0;
}
```

If you are done configuring Device R1, enter **commit** from configuration mode.

```
user@R2# show interfaces
ge-2/0/7 {
    description to-Host;
    unit 0 {
        family inet {
            address 172.16.80.2/30;
        }
    }
}
ge-2/0/8 {
    description to-R1;
    unit 0 {
        family inet {
            address 10.50.0.2/30;
        }
    }
}
lo0 {
    unit 0 {
        description looback-interface;
        family inet {
            address 192.168.14.1/32;
        }
    }
}
```

```
user@R2# show protocols ospf
area 0.0.0.0 {
    interface ge-2/0/7.0 {
        passive;
    }
    interface lo0.0 {
        passive;
    }
    interface ge-2/0/8.0;
```


Copyright © 2019, Juniper Networks, Inc.

```
}
```

If you are done configuring Device R2, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Clearing the Counters on page 141
- Sending TCP Traffic into the Network and Monitoring the Discards on page 141

Clearing the Counters

Purpose Confirm that the firewall counters are cleared.

Action On Device R1, run the **clear firewall all** command to reset the firewall counters to 0.

```
user@R1> clear firewall all
```

Sending TCP Traffic into the Network and Monitoring the Discards

Purpose Make sure that the traffic of interest that is sent is rate-limited on the output interface (ge-2/0/8).

Action

1. Use a traffic generator to send 20 TCP packets with a source port of 80.

   The -s flag sets the source port. The -k flag causes the source port to remain steady at 80 instead of incrementing. The -c flag sets the number of packets to 10. The -d flag sets the packet size.

   The destination IP address of 172.16.80.1 represents a user that is downstream of Device R2. The user has requested a webpage from the host (the webserver emulated by the traffic generator), and the packets are sent in response to the request.

NOTE: In this example the policer numbers are reduced to a bandwidth limit of 8 Kbps and a burst size limit of 1500 KBps to ensure that some packets are dropped.

```
[root@host]# hping 172.16.80.1 -s 80 -k -d 375 -c 20
```

```
[root@tp-lnx03 rtwright]# hping 172.16.80.1 -s 80 -k -d 375 -c 20
HPING 172.16.80.1 (eth1 172.16.80.1): NO FLAGS are set, 40 headers + 375 data
 bytes
len=46 ip=172.16.80.1 ttl=62 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=4000.8
ms
.
```

Copyright © 2019, Juniper Networks, Inc.

```
.
.
--- 172.16.80.1 hping statistic ---
20 packets transmitted, 12 packets received, 40% packet loss
```

2. On Device R1, check the firewall counters by using the **show firewall** command.

```
user@R1> show firewall
```

```
user@sugar# run show firewall

Filter: mf-classifier
Policers:
Name                                                Bytes              Packets
discard-t1                                  3320                8
```

Meaning In Steps 1 and 2 the output from both devices shows that 8 packets were discarded. This means that there was at least 8 Kbps of green (in-contract HTTP port 80) traffic and that the 1500 KBps burst option for red out-of-contract HTTP port 80 traffic was exceeded.

Related Documentation

- *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide*
- *Example: Configuring a Two-Rate Three-Color Policer*

Example: Limiting Inbound Traffic Within Your Network by Configuring an Ingress Single-Rate Two-Color Policer and Configuring Multifield Classifiers

This example shows how to limit customer traffic within your network using a single-rate two-color policer. Policers use a concept known as a token bucket to identify which traffic to drop. The policer enforces the class-of-service (CoS) strategy of in-contract and out-of-contract traffic at the interface level. You can apply a single-rate two-color policer to incoming packets, outgoing packets, or both. This example applies the policer as an input (ingress) policer for incoming traffic. The multifield classifier CoS queuing option places the traffic into the assigned queues which will help you manage resource utilization at the output interface level by applying scheduling and shaping at a later date.

A thorough explanation of the token bucket concept and its underlying algorithms is beyond the scope of this document. For more information about traffic policing, and CoS in general, refer to *QOS-Enabled Networks—Tools and Foundations* by Miguel Barreiros and Peter Lundqvist. This book is available at many online booksellers and at www.juniper.net/books .

- Requirements on page 143
- Overview on page 143

Copyright © 2019, Juniper Networks, Inc.

- Configuration on page 146
- Verification on page 151

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Overview

Policing

Single-rate two-color policing enforces a configured rate of traffic flow for a particular service level by applying implicit or configured actions to traffic that does not conform to the limits. When you apply a single-rate two-color policer to the input or output traffic at an interface, the policer meters the traffic flow to the rate limit defined by the following components:

- Bandwidth limit—The average number of bits per second permitted for packets received or transmitted at the interface. You can specify the bandwidth limit as an absolute number of bits per second or as a percentage value from 1 through 100. If a percentage value is specified, the effective bandwidth limit is calculated as a percentage of either the physical interface media rate or the logical interface configured shaping rate.
- Burst-size limit—The maximum size permitted for bursts of data. Burst sizes are measured in bytes. We recommend two formulas for calculating burst size:

  Burst size = bandwidth x allowable time for burst traffic / 8

  Or

  Burst size = interface mtu x 10

  For information about configuring the burst size, see *Determining Proper Burst Size for Traffic Policers*.

NOTE: **There is a finite buffer space for an interface. In general, the estimated total buffer depth for an interface is about 125 ms.**

For a traffic flow that conforms to the configured limits (categorized as green traffic), packets are implicitly marked with a packet loss priority (PLP) level of low and are allowed to pass through the interface unrestricted.

For a traffic flow that exceeds the configured limits (categorized as red traffic), packets are handled according to the traffic-policing actions configured for the policer. This example discards packets that burst over the 15 KBps limit.

To rate-limit Layer 3 traffic, you can apply a two-color policer in the following ways:

Copyright © 2019, Juniper Networks, Inc.

- Directly to a logical interface, at a specific protocol level.
- As the action of a standard stateless firewall filter that is applied to a logical interface, at a specific protocol level. This is the technique used in this example.

To rate-limit Layer 2 traffic, you can apply a two-color policer as a logical interface policer only. You cannot apply a two-color policer to Layer 2 traffic through a firewall filter.

CAUTION: You can choose either bandwidth-limit or bandwidth percent within the policer, as they are mutually exclusive. You cannot configure a policer to use bandwidth percent for aggregate, tunnel, or software interfaces.

In this example, the host is a traffic generator emulating a webserver. Devices R1 and R2 are owned by a service provider. The webserver is accessed by users behind Device R2. The host will be sending traffic with a source port TCP HTTP port 80 and a source port 12345 to the users. A single-rate two-color policer is configured and applied to the interface on Device R1 that connects the host to Device R1. The policer enforces the contractual bandwidth availability made between the owner of the webserver (in this case emulated by the host) and the service provider that owns Device R1 for the web traffic that flows over the link that connects the host to Device R1.

In accordance with the contractual bandwidth availability made between the owner of the webserver and the service provider that owns Devices R1 and R2, the policer will limit the HTTP port 80 traffic and the port 12345 traffic originating from the host to using 700 Mbps (70 percent) of the available bandwidth with an allowable burst rate of 10 x the MTU size of the gigabit Ethernet interface between the host and Device R1.

NOTE: In a real-world scenario you would probably also rate-limit traffic for a variety of other ports such as FTP, SFTP, SSH, TELNET, SMTP, IMAP, and POP3 because they are often included as additional services with web hosting services.

NOTE: You need to leave some additional bandwidth available that is not rate-limited for network control protocols such as routing protocols, DNS, and any other protocols required to keep network connectivity operational. This is why the firewall filter has a final accept condition on it.

Topology

This example uses the topology in Figure 17 on page 127.

Copyright © 2019, Juniper Networks, Inc.

Figure 21: Single-Rate Two-Color Policer Scenario

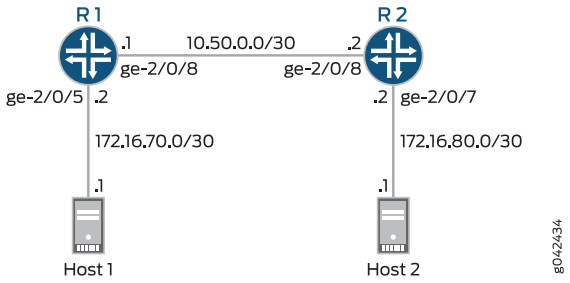
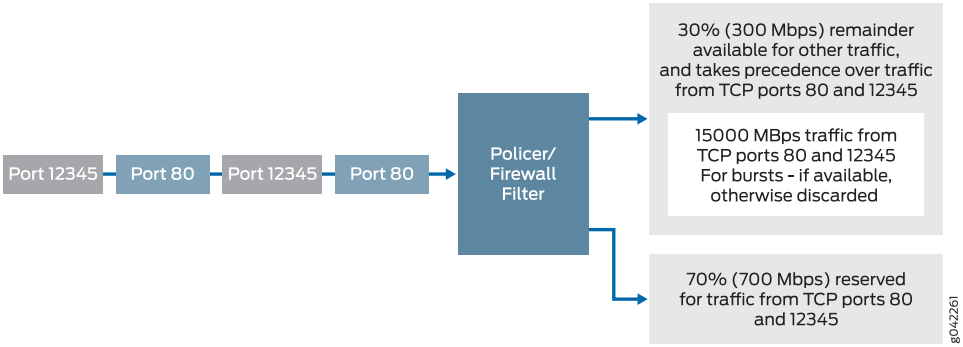


Figure 18 on page 127 shows the policing behavior.

Figure 22: Traffic Limiting in a Single-Rate Two-Color Policer Scenario



Multifield Classifying

A classifier is a software operation that a router or switch uses to inspect and classify a packet after it has made it through any policing, if policing is configured. During classification, the packet header contents are examined, and this examination determines how the packet is treated when the outbound interface becomes too busy to handle all of the packets and you want your device to drop packets intelligently, instead of dropping packets indiscriminately. One common way to detect packets of interest is by source port number. The TCP source port numbers 80 and 12345 are used in this example, but many other matching criteria for packet detection are available to multifield classifiers, using firewall filter match conditions. The configuration in this example specifies that TCP packets with a source port 80 are classified into the BE-data forwarding class and queue number 0, and TCP packets with a source port 12345 are classified into the Premium-data forwarding class and queue number 1. Traffic from both port numbers is monitored by the policer first. If the traffic makes it through the policer, it is handed off to the outbound interface in the assigned queue for transmission.

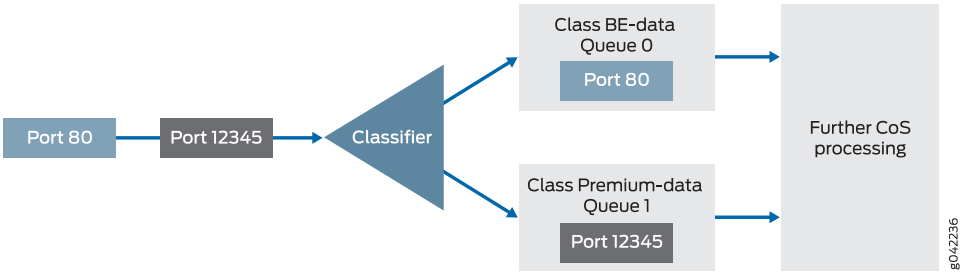
Multifield classifiers are typically used at the network edge as packets enter an autonomous system (AS).

Copyright © 2019, Juniper Networks, Inc.

In this example, you configure the firewall filter mf-classifier and specify some custom forwarding classes on Device R1. In specifying the custom forwarding classes, you also associate each class with a queue.

The classifier operation is shown in Figure 14 on page 104.

Figure 23: Multifield Classifier Based on TCP Source Ports



You apply the multifield classifier's firewall filter as an input filter on each customer-facing or host-facing interface that needs the filter. In this example, the incoming interface ge-2/0/5 on Device R1 is used. You monitor the behavior of the queues on the interfaces that the traffic is transmitted over. In this example, to determine how the queues are being serviced, you examine the traffic statistics on interface ge-2/0/8 by using the **extensive** option in the **show interfaces** command.

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.

Device R1

```
set interfaces ge-2/0/5 description to-Host
set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
set interfaces ge-2/0/5 unit 0 family inet filter input mf-classifier
set interfaces ge-2/0/8 description to-R2
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.13.1/32
set firewall policer discard if-exceeding bandwidth-limit 700m
set firewall policer discard if-exceeding burst-size-limit 15k
set firewall policer discard then discard
set class-of-service forwarding-classes class BE-data queue-num 0
set class-of-service forwarding-classes class Premium-data queue-num 1
set class-of-service forwarding-classes class Voice queue-num 2
set class-of-service forwarding-classes class NC queue-num 3
set firewall family inet filter mf-classifier term BE-data from protocol tcp
set firewall family inet filter mf-classifier term BE-data from port http
set firewall family inet filter mf-classifier term BE-data then forwarding-class BE-data
set firewall family inet filter mf-classifier term BE-data then policer discard
set firewall family inet filter mf-classifier term Premium-data from protocol tcp
set firewall family inet filter mf-classifier term Premium-data from port 12345
```


Copyright © 2019, Juniper Networks, Inc.

```
set firewall family inet filter mf-classifier term Premium-data then forwarding-class
  Premium-data
set firewall family inet filter mf-classifier term Premium-data then policer discard
set firewall family inet filter mf-classifier term accept then accept
set protocols ospf area 0.0.0.0 interface ge-2/0/5.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Device R2

```
set interfaces ge-2/0/7 description to-Host
set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
set interfaces ge-2/0/8 description to-R1
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.14.1/32
set protocols ospf area 0.0.0.0 interface ge-2/0/7.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Device R1:

1. Configure the device interfaces.

```
[edit interfaces]
user@R1#set ge-2/0/5 description to-Host
user@R1#set ge-2/0/5 unit 0 family inet address 172.16.70.2/30
user@R1#set ge-2/0/8 description to-R2
user@R1#set ge-2/0/8 unit 0 family inet address 10.50.0.1/30
user@R1# set lo0 unit 0 description looback-interface
user@R1#set lo0 unit 0 family inet address 192.168.13.1/32
```

2. Configure the policer to rate-limit to a bandwidth of 700 Mbps and a burst size of 15 KBps.

```
[edit firewall policer discard]
user@R1# set if-exceeding bandwidth-limit 700m
user@R1# set if-exceeding burst-size-limit 15k
```

3. Configure the policer to discard packets in the red traffic flow.

```
[edit firewall policer discard]
user@R1# set then discard
```

4. Configure the custom forwarding classes and associated queue numbers.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service forwarding-classes]
user@R1# set class BE-data queue-num 0
user@R1# set class Premium-data queue-num 1
user@R1# set class Voice queue-num 2
user@R1# set class NC queue-num 3
```

5. Configure the firewall filter term that places TCP traffic with a source port of 80 (HTTP traffic) into the BE-data forwarding class, associated with queue 0.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term BE-data from protocol tcp
user@R1# set term BE-data from port http
user@R1# set term BE-data then forwarding-class BE-data
user@R1# set term BE-data then policer discard
```

6. Configure the firewall filter term that places TCP traffic with a source port of 12345 into the Premium-data forwarding class, associated with queue 1.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term Premium-data from protocol tcp
user@R1# set term Premium-data from port 12345
user@R1# set term Premium-data then forwarding-class Premium-data
user@R1# set term Premium-data then policer discard
```

7. At the end of your firewall filter, configure a default term that accepts all other traffic.

   Otherwise, all traffic that arrives on the interface and is not explicitly accepted by the firewall filter is discarded.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term accept then accept
```

8. Apply the firewall filter to the ge-2/0/5 interface as an input filter.

```
[edit interfaces]
user@R1# set ge-2/0/5 unit 0 family inet filter input mf-classifier
```

9. Configure OSPF.

```
[edit protocols ospf]
user@R1# set area 0.0.0.0 interface ge-2/0/5.0 passive
user@R1# set area 0.0.0.0 interface lo0.0 passive
user@R1# set area 0.0.0.0 interface ge-2/0/8.0
```


Copyright © 2019, Juniper Networks, Inc.

Step-by-Step Procedure

To configure Device R2:

1. Configure the device interfaces.

```
[edit interfaces]
user@R2# set ge-2/0/7 description to-Host
user@R2# set ge-2/0/7 unit 0 family inet address 172.16.80.2/30
user@R2# set ge-2/0/8 description to-R1
user@R2# set ge-2/0/8 unit 0 family inet address 10.50.0.2/30
user@R2# set lo0 unit 0 description looback-interface
user@R2# set lo0 unit 0 family inet address 192.168.14.1/32
```

2. Configure OSPF.

```
[edit protocols ospf]
user@R2#  set area 0.0.0.0 interface ge-2/0/7.0 passive
user@R2#  set area 0.0.0.0 interface lo0.0 passive
user@R2#  set area 0.0.0.0 interface ge-2/0/8.0
```

Results

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

```
user@R1# show interfaces
ge-2/0/5 {
   description to-Host;
   unit 0 {
      family inet {
         filter {
            input mf-classifier;
         }
         address 172.16.70.2/30;
      }
   }
}
ge-2/0/8 {
   description to-R2;
   unit 0 {
      family inet {
         address 10.50.0.1/30;
      }
   }
}
lo0 {
   unit 0 {
      description looback-interface;
      family inet {
         address 192.168.13.1/32;
      }
   }
```

Copyright © 2019, Juniper Networks, Inc.

```
}
```

```
user@R1# show class-of-service
forwarding-classes {
   class BE-data queue-num 0;
   class Premium-data queue-num 1;
   class Voice queue-num 2;
   class NC queue-num 3;
}
```

```
user@R1# show firewall
family inet {
   filter mf-classifier {
      term BE-data {
         from {
            protocol tcp;
            port http;
         }
         then {
            policer discard;
            forwarding-class BE-data;
         }
      }
      term Premium-data {
         from {
            protocol tcp;
            port 12345;
         }
         then {
            policer discard;
            forwarding-class Premium-data;
         }
      }
      term accept {
         then accept;
      }
   }
}
policer discard {
   if-exceeding {
      bandwidth-limit 700m;
      burst-size-limit 15k;
   }
   then discard;
}
```

```
user@R1# show protocols ospf
area 0.0.0.0 {
   interface ge-2/0/5.0 {
      passive;
   }
   interface lo0.0 {
      passive;
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
    interface ge-2/0/8.0;
}
```

If you are done configuring Device R1, enter **commit** from configuration mode.

```
user@R2# show interfaces
ge-2/0/7 {
    description to-Host;
    unit 0 {
        family inet {
            address 172.16.80.2/30;
        }
    }
}
ge-2/0/8 {
    description to-R1;
    unit 0 {
        family inet {
            address 10.50.0.2/30;
        }
    }
}
lo0 {
    unit 0 {
        description looback-interface;
        family inet {
            address 192.168.14.1/32;
        }
    }
}
```

```
user@R2# show protocols ospf
area 0.0.0.0 {
    interface ge-2/0/7.0 {
        passive;
    }
    interface lo0.0 {
        passive;
    }
    interface ge-2/0/8.0;
}
```

If you are done configuring Device R2, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Checking the CoS Settings on page 152
- Clearing the Counters on page 152

Copyright © 2019, Juniper Networks, Inc.

- Sending Traffic into the Network from TCP HTTP Port 80 and Monitoring the Results on page 152
- Sending Traffic into the Network from TCP Port 12345 and Monitoring the Results on page 154

Checking the CoS Settings

Purpose Confirm that the forwarding classes are configured correctly.

Action From Device R1, run the **show class-of-service forwarding-class** command.

```
user@R1> show class-of-service forwarding-class

Forwarding class                          ID      Queue  Restricted queue  Fabric
priority  Policing priority   SPU priority
  BE-data                                 0       0          0              low
            normal             low
  Premium-data                            1       1          1              low
            normal             low
  Voice                                   2       2          2              low
            normal             low
  NC                                      3       3          3              low
            normal             low
```

Meaning The output shows the configured custom classifier settings.

Clearing the Counters

Purpose Confirm that the firewall and interface counters are cleared.

Action

- On Device R1, run the **clear firewall all** command to reset the firewall counters to 0.

  ```
  user@R1> clear firewall all
  ```

- On Device R1, run the **clear interface statistics ge-2/0/5** command to reset the interface counters to 0.

  ```
  user@R1> clear interface statistics ge-2/0/8
  ```

Sending Traffic into the Network from TCP HTTP Port 80 and Monitoring the Results

Purpose Send traffic that can monitored at the policer and custom queue level.

Action

1. Use a traffic generator to send 20 TCP packets with a source port of 80 into the network.

Copyright © 2019, Juniper Networks, Inc.

The -s flag sets the source port. The -k flag causes the source port to remain steady at 80 instead of incrementing. The -c flag sets the number of packets to 20. The -d flag sets the packet size.

NOTE: In this example the policer numbers are reduced to a bandwidth limit of 8 Kbps and a burst size limit of 1500 KBps to ensure that some packets are dropped.

```
[User@host]#  hping 172.16.80.1 -c 20 -s 80 -k -d 300
```

```
[root@host]# hping 172.16.80.1 -s 80 -k -c 20 -d 300
HPING 172.16.80.1 (eth1 172.16.80.1): NO FLAGS are set, 40 headers + 300 data
 bytes
len=46 ip=172.16.80.1 ttl=62 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=1.4 ms
.
.
.
--- 172.16.80.1 hping statistic ---
20 packets transmitted, 16 packets received, 20% packet loss
round-trip min/avg/max = 1.4/8688.9/17002.3 ms
```

2. On Device R1, check the firewall counters by using the **show firewall** command.

```
user@R1>  show firewall
```

```
Filter: mf-classifier
Policers:
Name                                                Bytes              Packets
discard-BE-data                                      1360                    4
discard-Premium-data                                    0                    0
```

Notice that in the hping output that there was 20% packet loss (4 packets out of 20) and the same number of packets were dropped by the policer as shown in the output of the **show firewall** command. Also notice that the drops are associated with the queue BE-data as specified in the mf-classifier in the firewall configuration.

3. On Device R1, check the queue counters by using the **show interfaces extensive ge-2/0/8| find "Queue counters"** command.

```
user@R1>  show interfaces extensive ge-2/0/8| find "Queue counters"
```

```
 Queue counters:      Queued packets  Transmitted packets     Dropped packets

   0                                16                   16                   0
   1                                 0                    0                   0
   2                                 0                    0                   0
   3                                 4                    4                   0
 Queue number:         Mapped forwarding classes
```

Copyright © 2019, Juniper Networks, Inc.

```
0                   BE-data
1                   Premium-data
2                   Voice
3                   NC
```

Notice that 16 packets were transmitted out interface 2/0/8 using the queue BE-data as specified in the mf-classifier in the firewall configuration. The remaining 4 packets, were dropped by the policer, as shown above. The 4 packets sent to queue 3 are network control traffic. They are possibly routing protocol updates.

Meaning The output from both devices shows that 4 packets were discarded This means that there was at least 8 Kbps of green (in-contract HTTP port 80) traffic and that the 1500 KBps burst option for red out-of-contract HTTP port 80 traffic was exceeded. In Steps 2 and 3, you can see that the correct queues were used to transmit the remaining traffic out interface 2/0/8.

Sending Traffic into the Network from TCP Port 12345 and Monitoring the Results

Purpose Send traffic that can monitored at the policer and custom queue level.

Action

1. Clear the counters again as shown in section "Clearing the Counters" on page 152.
2. Use a traffic generator to send 20 TCP packets with a source port of 12345 into the network.

   The -s flag sets the source port. The -k flag causes the source port to remain steady at 12345 instead of incrementing. The -c flag sets the number of packets to 20. The -d flag sets the packet size.

   ```
   [User@host]#  hping 172.16.80.1 -c 20 -s 12345 -k -d 300
   ```

   ```
   [root@tp-host]# hping 172.16.80.1 -s 12345 -k -c 20 -d 300
   HPING 172.16.80.1 (eth1 172.16.80.1): NO FLAGS are set, 40 headers + 300 data
    bytes
   len=46 ip=172.16.80.1 ttl=62 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=0.4 ms
   .
   .
   .
   --- 172.16.80.1 hping statistic ---
   20 packets transmitted, 16 packets received, 20% packet loss
   round-trip min/avg/max = 0.4/9126.3/18002.4 ms
   ```

3. On Device R1, check the firewall counters by using the **show firewall** command.

   ```
   user@R1> show firewall
   ```

   ```
   Filter: mf-classifier
   Policers:
   Name                                                Bytes              Packets
   discard-BE-data                                         0                    0
   ```


Copyright © 2019, Juniper Networks, Inc.

```
discard-Premium-data                                  1360                4
```

Notice that in the hping output that there was 20% packet loss (4 packets out of 20) and the same number of packets were dropped by the policer as shown in the output of the **show firewall** command. Also notice that the drops are associated with the queue Premium-data as specified in the mf-classifier in the firewall configuration.

4. On Device R1, check the queue counters by using the **show interfaces extensive ge-2/0/8| find "Queue counters"** command.

```
user@R1> show interfaces extensive ge-2/0/8| find "Queue counters"

Queue counters:      Queued packets  Transmitted packets     Dropped packets
  0                                0                   0                   0
  1                               16                  16                   0
  2                                0                   0                   0
  3                               19                  19                   0
Queue number:         Mapped forwarding classes
  0                   BE-data
  1                   Premium-data
  2                   Voice
  3                   NC
```

Notice that 16 packets were transmitted out interface 2/0/8 using the Premium-data queues as specified in the mf-classifier firewall configuration. The remaining 4 packets were dropped by the policer, as shown above. The 19 packets sent to queue 3 are network control traffic. They are possibly routing protocol updates.

Meaning The output from both devices shows that 4 packets were discarded. This means that there was at least 8 Kbps of green (in-contract HTTP port 80) traffic and that the 1500 KBps burst option for red out-of-contract HTTP port 80 traffic was exceeded. In Steps 3 and 4, you can see that the correct queues were used to transmit the remaining traffic out interface 2/0/8.

Related Documentation

- *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide*
- *Example: Configuring a Two-Rate Three-Color Policer*

Example: Limiting Outbound Traffic Within Your Network by Configuring an Egress Single-Rate Two-Color Policer and Configuring Multifield Classifiers

This example shows how to limit customer traffic within your network using a single-rate two-color policer. Policers use a concept known as a token bucket to identify which traffic to drop. The policer enforces the class-of-service (CoS) strategy of in-contract and out-of-contract traffic at the interface level. You can apply a single-rate two-color policer to incoming packets, outgoing packets, or both. This example applies the policer as an output (egress) policer for outgoing traffic. The multifield classifier CoS queueing option

Copyright © 2019, Juniper Networks, Inc.

places the traffic into the assigned queues which will help you manage resource utilization at the output interface level by applying scheduling and shaping later.

A thorough explanation of the token bucket concept and its underlying algorithms is beyond the scope of this document. For more information about traffic policing, and CoS in general, refer to *QOS-Enabled Networks—Tools and Foundations* by Miguel Barreiros and Peter Lundqvist. This book is available at many online booksellers and at www.juniper.net/books.

- Requirements on page 156
- Overview on page 156
- Configuration on page 160
- Verification on page 167

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Overview

Policing

Single-rate two-color policing enforces a configured rate of traffic flow for a particular service level by applying implicit or configured actions to traffic that does not conform to the limits. When you apply a single-rate two-color policer to the input or output traffic at an interface, the policer meters the traffic flow to the rate limit defined by the following components:

- Bandwidth limit—The average number of bits per second permitted for packets received or transmitted at the interface. You can specify the bandwidth limit as an absolute number of bits per second or as a percentage value from 1 through 100. If a percentage value is specified, the effective bandwidth limit is calculated as a percentage of either the physical interface media rate or the logical interface configured shaping rate.
- Burst-size limit—The maximum size permitted for bursts of data. Burst sizes are measured in bytes. We recommend two formulas for calculating burst size:

  Burst size = bandwidth x allowable time for burst traffic / 8

  Or

  Burst size = interface mtu x 10

  For information about configuring the burst size, see *Determining Proper Burst Size for Traffic Policers*.

Copyright © 2019, Juniper Networks, Inc.

NOTE: There is a finite buffer space for an interface. In general, the estimated total buffer depth for an interface is about 125 ms.

For a traffic flow that conforms to the configured limits (categorized as green traffic), packets are implicitly marked with a packet loss priority (PLP) level of **low** and are allowed to pass through the interface unrestricted.

For a traffic flow that exceeds the configured limits (categorized as red traffic), packets are handled according to the traffic-policing actions configured for the policer. This example discards packets that burst over the 15 KBps limit.

To rate-limit Layer 3 traffic, you can apply a two-color policer in the following ways:

- Directly to a logical interface, at a specific protocol level.
- As the action of a standard stateless firewall filter that is applied to a logical interface, at a specific protocol level. This is the technique used in this example.

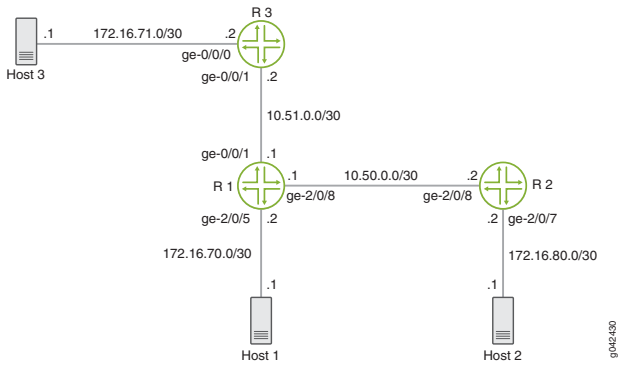
To rate-limit Layer 2 traffic, you can apply a two-color policer as a logical interface policer only. You cannot apply a two-color policer to Layer 2 traffic through a firewall filter.

CAUTION: You can choose either bandwidth-limit or bandwidth percent within the policer, as they are mutually exclusive. You cannot configure a policer to use bandwidth percent for aggregate, tunnel, and software interfaces.

In this example, as illustrated in Figure 17 on page 127, Host1 connected to device R1 and Host3 connected to device R3 are traffic generators emulating webservers. Both Host1 and Host3 are sending traffic to Host2 behind device R2. Devices R1, R2, and R3 are owned by a service provider. Host1 is accessed by users on Host2 behind R2. Host1 and Host2 are owned by the same customer and their traffic must be managed. Host1 will be sending traffic with a source TCP HTTP port of 80 to the users. A single-rate two-color policer is configured and applied to the interface on R1 that connects to R2. The policer enforces the contract agreed upon by the webserver owner and the service provider for the bandwidth available to the web traffic flowing between R1 and R2.

Copyright © 2019, Juniper Networks, Inc.

Figure 24: Single-Rate Two-Color Policer Scenario



This example applies an egress policer between R1 and R2 because this is the point where the traffic from both customer sites shares the same link. This makes it easier to enforce the required policing parameters. Trying to rate-limit the combined customer traffic on the link between R1 and R2 by applying the policers as ingress policers on interfaces ge-0/0/0 on R3 and ge-2/0/5 on R1 would be complicated because using the contracted rate of 700 Mbps (70 percent) of the available bandwidth with an allowable burst rate of 10 x the MTU size of the gigabit Ethernet interface between Host3 and R3 and the Host1 and R1 would result in allowing a maximum throughput of 1400 Mbps over the link between R1 and R2.

Therefore, the rate-limiting applied to the host connections between the hosts and R3 and R1 would have to be reduced below 700 Mbps. The calculation of what to reduce the rate-limit number to would be a problem because just reducing each host to 350 Mbps would mean that if one host was transmitting traffic while the other host was not transmitting, the maximum throughput on the link between R1 and R2 would be only one half of the contracted rate (350 Mbps instead of 700 Mbps). This is why this example is useful to show the amount of thought that must go into applying CoS in a network to achieve the desired goals.

According to the contractual bandwidth availability, the egress policer on R1 will limit the HTTP port 80 traffic originating from Host1 to using 700 Mbps (70 percent) of the available bandwidth with an allowable burst rate of 10 x the MTU size of the gigabit Ethernet interface between R1 and R2.

Additional traffic from TCP source port 12345 is used in this example to further illustrate how traffic is allocated to the outbound queues.

NOTE: In a real-world scenario you would probably also rate-limit traffic for a variety of other ports such as FTP, SFTP, SSH, TELNET, SMTP, IMAP, and POP3 because they are often included as additional services with web hosting services.

Copyright © 2019, Juniper Networks, Inc.

NOTE: You must leave some additional bandwidth available that is not rate-limited for network control protocols such as routing protocols, DNS, and any other protocols required to keep network connectivity operational. This is why the firewall filter has a final accept condition on it.

Topology

This example uses the topology in Figure 18 on page 127.

Figure 25: Single-Rate Two-Color Policer Scenario

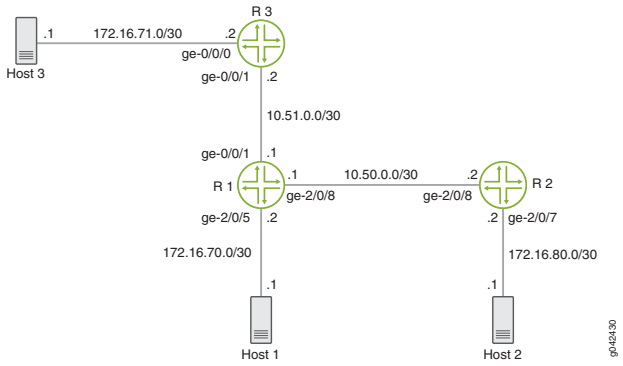
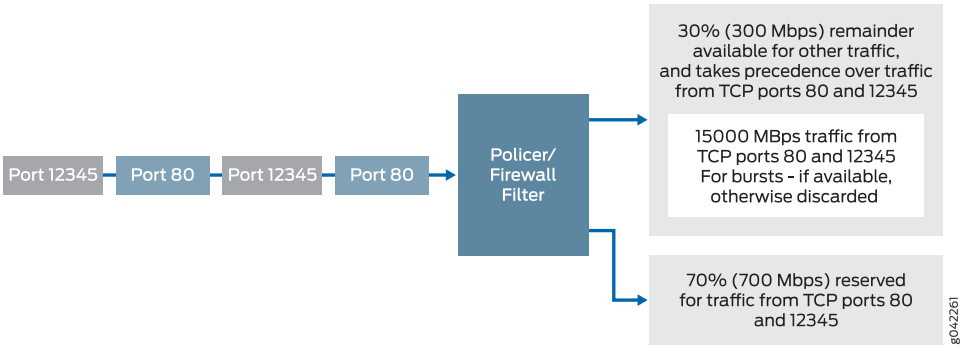


Figure 26 on page 159 shows the policing behavior.

Figure 26: Traffic Limiting in a Single-Rate Two-Color Policer Scenario



Multifield Classifying

A classifier is a software operation that a router or switch uses to inspect and classify a packet after it has made it through any policing, if policing is configured. During classification, the packet header contents are examined, and this examination determines how the packet is treated when the outbound interface becomes too busy to handle all of the packets and you want your device to drop packets intelligently, instead of dropping packets indiscriminately. One common way to detect packets of interest is by source port number. The TCP source port numbers 80 and 12345 are used in this example, but many other matching criteria for packet detection are available to multifield classifiers,

Copyright © 2019, Juniper Networks, Inc.

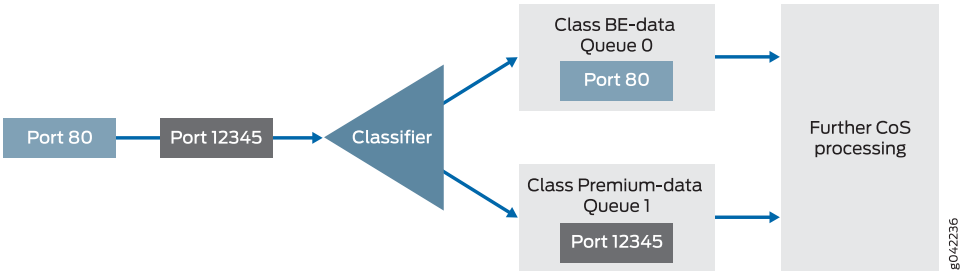
using firewall filter match conditions. The configuration in this example specifies that TCP packets with a source port 80 are classified into the BE-data forwarding class and queue number 0, and TCP packets with a source port 12345 are classified into the Premium-data forwarding class and queue number 1. Traffic from both port numbers is monitored by the policer first. If the traffic makes it through the policer, it is handed off to the outbound interface in the assigned queue for transmission.

Multifield classifiers are typically used at the network edge as packets enter an autonomous system (AS). However, as explained previously in the policing section, in this example the multifield classifier is configured within the AS of the service provider.

In this example, you configure the firewall filter **mf-classifier** and specify some custom forwarding classes on R1. In specifying the custom forwarding classes, you also associate each class with a queue.

The classifier operation is shown in Figure 14 on page 104.

Figure 27: Multifield Classifier Based on TCP Source Ports



You monitor the behavior of the queues on the interfaces that the traffic is transmitted over. In this example, to determine how the queues are being serviced, you examine the traffic statistics on interface ge-2/0/8 on R1 by using the **extensive** option in the **show interfaces** command.

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.

Device R1

```
set interfaces ge-0/0/1 description to-R3
set interfaces ge-0/0/1 unit 0 family inet address 10.51.0.1/30
set interfaces ge-2/0/5 description to-Host
set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
set interfaces ge-2/0/8 description to-R2
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
set interfaces ge-2/0/8 unit 0 family inet filter output mf-classifier
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 192.168.13.1/32
set firewall policer discard if-exceeding bandwidth-limit 700m
set firewall policer discard if-exceeding burst-size-limit 15k
set firewall policer discard then discard
```


Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service forwarding-classes class BE-data queue-num 0
set class-of-service forwarding-classes class Premium-data queue-num 1
set class-of-service forwarding-classes class Voice queue-num 2
set class-of-service forwarding-classes class NC queue-num 3
set firewall family inet filter mf-classifier term BE-data from protocol tcp
set firewall family inet filter mf-classifier term BE-data from port http
set firewall family inet filter mf-classifier term BE-data then forwarding-class BE-data
set firewall family inet filter mf-classifier term BE-data then policer discard
set firewall family inet filter mf-classifier term Premium-data from protocol tcp
set firewall family inet filter mf-classifier term Premium-data from port 12345
set firewall family inet filter mf-classifier term Premium-data then forwarding-class
  Premium-data
set firewall family inet filter mf-classifier term Premium-data then policer discard
set firewall family inet filter mf-classifier term accept then accept
set protocols ospf area 0.0.0.0 interface ge-2/0/5.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-0/0/1.0
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Device R2

```
set interfaces ge-2/0/7 description to-Host
set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
set interfaces ge-2/0/8 description to-R1
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 192.168.14.1/32
set protocols ospf area 0.0.0.0 interface ge-2/0/7.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Device R3

```
set interfaces ge-0/0/0 description to-Host
set interfaces ge-0/0/0 unit 0 family inet address 172.16.71.1/30
set interfaces ge-0/0/1 description to-R1
set interfaces ge-0/0/1 unit 0 family inet address 10.51.0.2/30
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 192.168.15.1/32
set protocols ospf area 0.0.0.0 interface ge-0/0/0.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-0/0/1.0
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure R1:

1. Configure the device interfaces.

```
[edit interfaces]
user@R1# set ge-0/0/1 description to-R3
user@R1# set ge-0/0/1 unit 0 family inet address 10.51.0.1/30
```

Copyright © 2019, Juniper Networks, Inc.

```
user@R1# set ge-2/0/5 description to-Host
user@R1# set ge-2/0/5 unit 0 family inet address 172.16.70.2/30
user@R1# set ge-2/0/8 description to-R2
user@R1# set ge-2/0/8 unit 0 family inet address 10.50.0.1/30
```

2. Configure the policer to rate-limit to a bandwidth of 700 Mbps and a burst size of 15 KBps.

```
[edit firewall policer discard]
user@R1# set if-exceeding bandwidth-limit 700m
user@R1# set if-exceeding burst-size-limit 15k
```

3. Configure the policer to discard packets in the red traffic flow.

```
[edit firewall policer discard]
user@R1# set then discard
```

4. Configure the custom forwarding classes and associated queue numbers.

```
[edit class-of-service forwarding-classes]
user@R1# set class BE-data queue-num 0
user@R1# set class Premium-data queue-num 1
user@R1# set class Voice queue-num 2
user@R1# set class NC queue-num 3
```

5. Configure the firewall filter term that places TCP traffic with a source port of 80 (HTTP traffic) into the BE-data forwarding class, associated with queue 0.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term BE-data from protocol tcp
user@R1# set term BE-data from port http
user@R1# set term BE-data then forwarding-class BE-data
user@R1# set term BE-data then policer discard
```

6. Configure the firewall filter term that places TCP traffic with a source port of 12345 into the Premium-data forwarding class, associated with queue 1.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term Premium-data from protocol tcp
user@R1# set term Premium-data from port 12345
user@R1# set term Premium-data then forwarding-class Premium-data
user@R1# set term Premium-data then policer discard
```

7. At the end of your firewall filter, configure a default term that accepts all other traffic.

   Otherwise, all traffic that arrives on the interface that is not explicitly accepted by the firewall filter is discarded.

Copyright © 2019, Juniper Networks, Inc.

```
[edit firewall family inet filter mf-classifier]
user@R1# set term accept then accept
```

8. Apply the firewall filter to interface ge-2/0/8 as an output filter.

```
[edit interfaces]
user@R1# set ge-2/0/8 unit 0 family inet filter output mf-classifier
```

9. Configure OSPF.

```
[edit protocols ospf]
user@R1# set area 0.0.0.0 interface ge-2/0/5.0 passive
user@R1# set area 0.0.0.0 interface lo0.0 passive
user@R1# set area 0.0.0.0 interface ge-0/0/1.0
user@R1# set area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

To configure R2:

1. Configure the device interfaces.

```
[edit]
user@R2# set interfaces ge-2/0/7 description to-Host
user@R2# set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
user@R2# set interfaces ge-2/0/8 description to-R1
user@R2# set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
user@R2# set interfaces lo0 unit 0 description loopback-interface
user@R2# set interfaces lo0 unit 0 family inet address 192.168.14.1/32
```

Configure OSPF.

```
[edit protocols ospf]
user@R2# set area 0.0.0.0 interface ge-2/0/7.0 passive
user@R2# set area 0.0.0.0 interface lo0.0 passive
user@R2# set area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

To configure R3:

1. Configure the interfaces.

```
[edit]
user@R3# set interfaces ge-0/0/0 description to-Host
user@R3# set interfaces ge-0/0/0 unit 0 family inet address 172.16.71.1/30
user@R3# set interfaces ge-0/0/1 description to-R1
user@R3# set interfaces ge-0/0/1 unit 0 family inet address 10.51.0.2/30
user@R3# set interfaces lo0 unit 0 description loopback-interface
user@R3# set interfaces lo0 unit 0 family inet address 192.168.15.1/32
```

Copyright © 2019, Juniper Networks, Inc.

2. Configure OSPF

```
[edit protocols ospf]
user@R3# set protocols ospf area 0.0.0.0 interface ge-0/0/0.0 passive
user@R3# set protocols ospf area 0.0.0.0 interface lo0.0 passive
user@R3# set protocols ospf area 0.0.0.0 interface ge-0/0/1.0
```

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

```
user@R1# show interfaces
ge-0/0/1 {
   description to-R3;
   unit 0 {
      family inet {
         address 10.51.0.1/30;
      }
   }
}
}
   ge-2/0/5 {
      description to-Host;
      unit 0 {
         family inet {
            address 172.16.70.2/30;
         }
      }
   }
   ge-2/0/8 {
      description to-R2;
      unit 0 {
         family inet {
            filter {
               output mf-classifier;
            }
            address 10.50.0.1/30;
         }
      }
   }
lo0 {
   unit 0 {
      description loopback-interface;
      family inet {
         address 192.168.13.1/32;
      }
   }
}
```

```
user@R1# show class-of-service
forwarding-classes {
```


Copyright © 2019, Juniper Networks, Inc.

```
    class BE-data queue-num 0;
    class Premium-data queue-num 1;
    class Voice queue-num 2;
    class NC queue-num 3;
}
```

```
user@R1# show firewall
family inet {
    filter mf-classifier {
        term BE-data {
            from {
                protocol tcp;
                port http;
            }
            then {
                policer discard;
                forwarding-class BE-data;
            }
        }
        term Premium-data {
            from {
                protocol tcp;
                port 12345;
            }
            then {
                policer discard;
                forwarding-class Premium-data;
            }
        }
        term accept {
            then accept;
        }
    }
}
policer discard {
    if-exceeding {
        bandwidth-limit 700m;
        burst-size-limit 15k;
    }
    then discard;
}
```

```
user@R1# show protocols ospf
area 0.0.0.0 {
    interface ge-2/0/5.0 {
        passive;
    }
    interface lo0.0 {
        passive;
    }
    interface ge-0/0/1.0;
    interface ge-2/0/8.0;
}
```

Copyright © 2019, Juniper Networks, Inc.

If you are done configuring R1, enter **commit** from configuration mode.

```
user@R2# show interfaces
ge-2/0/7 {
   description to-Host;
   unit 0 {
      family inet {
         address 172.16.80.2/30;
      }
   }
}
ge-2/0/8 {
   description to-R1;
   unit 0 {
      family inet {
         address 10.50.0.2/30;
      }
   }
}
lo0 {
   unit 0 {
      description loopback-interface;
      family inet {
         address 192.168.14.1/32;
      }
   }
}
```

```
user@R2# show protocols ospf
area 0.0.0.0 {
   interface ge-2/0/7.0 {
      passive;
   }
   interface lo0.0 {
      passive;
   }
   interface ge-2/0/8.0;
}
```

If you are done configuring R2, enter **commit** from configuration mode.

```
user@R3# show interfaces
ge-0/0/0 {
   description to-Host;
   unit 0 {
      family inet {
         address 172.16.71.2/30;
      }
   }
}
ge-0/0/1 {
   description to-R1;
   unit 0 {
      family inet {
```


Copyright © 2019, Juniper Networks, Inc.

```
        address 10.51.0.2/30;
      }
    }
  }
  lo0 {
    unit 0 {
      description loopback-interface;
      family inet {
        address 192.168.15.1/32;
      }
    }
  }
```

```
user@R3# show protocols ospf
area 0.0.0.0 {
  interface ge-0/0/0.0 {
    passive;
  }
  interface lo0.0 {
    passive;
  }
  interface ge-0/0/1.0;
}
```

If you are done configuring R3, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Checking the CoS Settings on page 167
- Clearing the Counters on page 168
- Sending Traffic into the Network from TCP HTTP Port 80 and Monitoring the Results on page 168
- Sending Traffic into the Network from TCP Port 12345 and Monitoring the Results on page 169

Checking the CoS Settings

Purpose Confirm that the forwarding classes are configured correctly.

Action From R1, run the **show class-of-service forwarding-class** command.

```
user@R1> show class-of-service forwarding-class

Forwarding class                          ID      Queue  Restricted queue  Fabric
priority  Policing priority   SPU priority
  BE-data                                  0      0          0             low
            normal             low
  Premium-data                             1      1          1             low
            normal             low
```

```
Voice                                    2       2          2             low
          normal            low
NC                                       3       3          3             low
          normal            low
```

Meaning The output shows the configured custom classifier settings.

Clearing the Counters

Purpose Confirm that the firewall and interface counters are cleared.

Action

- On R1, run the **clear firewall all** command to reset the firewall counters to 0.

  `user@R1>` **clear firewall all**

- On R1, run the **clear interface statistics ge-2/0/5** command to reset the interface counters to 0.

  `user@R1>` **clear interface statistics ge-2/0/8**

Sending Traffic into the Network from TCP HTTP Port 80 and Monitoring the Results

Purpose Send traffic that can monitored at the policer and custom queue level.

Action

1. Use a traffic generator to send 20 TCP packets with a source port of 80 into the network.

   The **-s** flag sets the source port. The **-k** flag causes the source port to remain steady at 80 instead of incrementing. The **-c** flag sets the number of packets to 20. The **-d** flag sets the packet size.

   NOTE: In this example the policer numbers are reduced to a bandwidth limit of 8 Kbps and a burst size limit of 1500 KBps to ensure that some packets are dropped.

   `[User@host]#` **hping 172.16.80.1 -c 20 -s 80 -k -d 300**

```
[User@Host]# hping 172.16.80.1 -s 80 -k -c 20 -d 375
HPING 172.16.80.1 (eth1 172.16.80.1): NO FLAGS are set, 40 headers + 375 data
 bytes
len=46 ip=172.16.80.1 ttl=62 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=1001.0
ms
.
.
```


Copyright © 2019, Juniper Networks, Inc.

```
.
--- 172.16.80.1 hping statistic ---
20 packets transmitted, 14 packets received, 30% packet loss
round-trip min/avg/max = 1001.0/10287.1/19002.1 ms
```

2. On R1, check the firewall counters by using the **show firewall** command.

```
user@R1> show firewall

Filter: mf-classifier
Policers:
Name                                                Bytes              Packets
discard-BE-data                                      2490                    6
discard-Premium-data                                    0                    0
```

Notice that in the **hping** output that there was 30% packet loss (6 packets out of 20) and the same number of packets was dropped by the policer as shown in the output of the **show firewall** command. Also notice that the drops are associated with the queue **BE-data** as specified in the **mf-classifier** in the firewall configuration.

3. On R1, check the queue counters by using the **show interfaces extensive ge-2/0/8| find "Queue counters"** command.

```
user@R1> show interfaces extensive ge-2/0/8| find "Queue counters"
Queue counters:       Queued packets  Transmitted packets     Dropped packets

0                                 14                  14                   0
1                                  0                   0                   0
2                                  0                   0                   0
3                                 16                  16                   0
Queue number:       Mapped forwarding classes
0                   BE-data
1                   Premium-data
2                   Voice
3                   NC
```

Notice that 14 packets were transmitted out interface 2/0/8 using the queue **BE-data** as specified in the **mf-classifier** in the firewall configuration. The remaining 6 packets were dropped by the policer, as shown above. The 16 packets sent to queue 3 are network control traffic. They are possibly routing protocol updates.

Meaning The output from both devices shows that 6 packets were discarded This means that there was at least 8 Kbps of green (in-contract HTTP port 80) traffic and that the 1500 KBps burst option for red (out-of-contract HTTP port 80) traffic was exceeded. In Steps 2 and 3, you can see that the correct queues were used to transmit the remaining traffic out interface 2/0/8.

Sending Traffic into the Network from TCP Port 12345 and Monitoring the Results

Purpose Send traffic that can monitored at the policer and custom queue level.

Copyright © 2019, Juniper Networks, Inc.

Action

1. Clear the counters again as shown in section "Clearing the Counters" on page 152.
2. Use a traffic generator to send 20 TCP packets with a source port of 12345 into the network.

   The **-s** flag sets the source port. The **-k** flag causes the source port to remain steady at 12345 instead of incrementing. The **-c** flag sets the number of packets to 20. The **-d** flag sets the packet size.

   ```
   [User@host]#  hping 172.16.80.1 -c 20 -s 12345 -k -d 300
   ```

   ```
   [Host@User]# hping 172.16.80.1 -s 12345 -k -c 20 -d 375
   HPING 172.16.80.1 (eth1 172.16.80.1): NO FLAGS are set, 40 headers + 375 data
    bytes
   len=46 ip=172.16.80.1 ttl=62 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=1000.4
   ms
   .
   .
   .

   --- 172.16.80.1 hping statistic ---
   20 packets transmitted, 13 packets received, 35% packet loss
   round-trip min/avg/max = 1000.4/10924.5/19002.2 ms
   ```

3. On R1, check the firewall counters by using the **show firewall** command.

   ```
   user@R1> show firewall
   ```

   ```
   Filter: mf-classifier
   Policers:
   Name                                                Bytes              Packets
   discard-BE-data                                         0                  0
   discard-Premium-data                                 2905                  7
   ```

   Notice that in the **hping** output that there was 35% packet loss (7 packets out of 20) and the same number of packets were dropped by the policer as shown in the output of the **show firewall** command. Also notice that the drops are associated with the queue **Premium-data** as specified in the **mf-classifier** in the firewall configuration.

4. On R1, check the queue counters by using the **show interfaces extensive ge-2/0/8| find "Queue counters"** command.

   ```
   user@R1> show interfaces extensive ge-2/0/8| find "Queue counters"
   ```

   ```
   Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0                              0                  0                    0
    1                             13                 13                    0
    2                              0                  0                    0
    3                             16                 16                    0
   Queue number:       Mapped forwarding classes
   0                   BE-data
   1                   Premium-data
   2                   Voice
   3                   NC
   ```

   Notice that 13 packets were transmitted out interface 2/0/8 using the Premium-data queues specified in the **mf-classifier** in the firewall configuration. The remaining 7

Copyright © 2019, Juniper Networks, Inc.

packets were dropped by the policer, as shown above. The 16 packets sent to queue 3 are network control traffic. They are possibly routing protocol updates.

Meaning The output from both devices shows that 7 packets were discarded. This means that there was at least 8 Kbps of green (in-contract HTTP port 80) traffic and that the 1500 KBps burst option for red (out-of-contract HTTP port 80) traffic was exceeded. In Steps 3 and 4, you can see that the correct queues were used to transmit the remaining traffic out interface 2/0/8.

Related Documentation

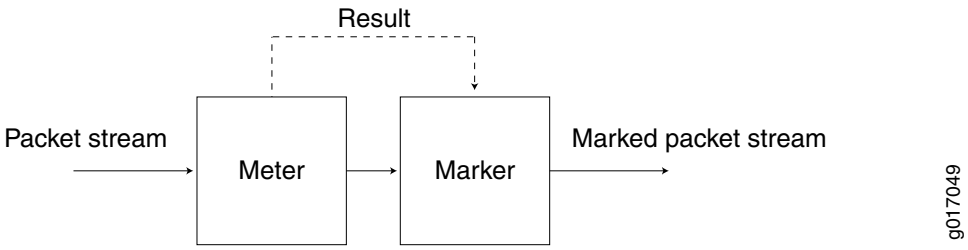
- *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide*
- *Example: Configuring a Two-Rate Three-Color Policer*

Overview of Tricolor Marking Architecture

Policers provide two functions: metering and marking.

The policer meters each packet and passes the packet and the metering result to the marker, as shown in Figure 28 on page 171.

Figure 28: Flow of Tricolor Marking Policer Operation



The meter operates in two modes. In the color-blind mode, the meter treats the packet stream as uncolored. Any preset loss priorities are ignored. In the color-aware mode, the meter inspects the packet loss priority (PLP) field, which has been set by an upstream device as PLP high, medium-high, medium-low, or low; in other words, the PLP field has already been set by a behavior aggregate (BA) or multifield classifier. The marker changes the PLP of each incoming IP packet according to the results of the meter. For more information, see "Configuring Two-Rate Tricolor Marking" on page 184.

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

Copyright © 2019, Juniper Networks, Inc.

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

For information about how to use marking policers with BA and multifield classifiers, see "Configuring Behavior Aggregate Classifiers" on page 51 and "Using Multifield Classifiers to Set Packet Loss Priority" on page 101.

Related Documentation

- Enabling Tricolor Marking and Limitations of Three-Color Policers on page 173
- Configuring and Applying Tricolor Marking Policers on page 175

Copyright © 2019, Juniper Networks, Inc.

Enabling Tricolor Marking and Limitations of Three-Color Policers

Copyright © 2019, Juniper Networks, Inc.

This topic describes how to enable TCM on Juniper Networks devices, as well as limitations you need to be aware of when you are using TCM.

Table 17 on page 174 lists the default state for TCM on Juniper Networks devices:

Table 17: Devices Versus TCM

TCM Enabled by Default	TCM Disabled by Default
M120 routers	M320 routers with Enhanced II FPCs
MX Series routers	T Series routers with Enhanced II FPCs
T4000 routers	T640 routers with Enhanced Scaling FPC4s
EX Series switches	T1600 routers with Enhanced Scaling FPC4s

NOTE: If you do not enable TCM on platforms that require it, you cannot configure **medium-low** or **medium-high** packet loss priority (PLP) for classifiers, rewrite rules, drop profiles, or firewall filters.

NOTE: On MX Series and M120 routers, you can apply three-color policers to aggregated interfaces.

NOTE: On T Series routers, three-color policers and hierarchical policers are supported on aggregated interfaces if all child links are hosted on Enhanced Scaling FPCs.

TCM has some limitations that must be kept in mind during configuration and operation.

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

Copyright © 2019, Juniper Networks, Inc.

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

- When some PICs with Frame Relay encapsulation mark a packet with high loss priority, the packet is treated as having medium-high loss priority on M320 Multiservice Edge Routers and T Series Core Routers with Enhanced II FPCs and T640 Core Routers with Enhanced Scaling FPC4.
- In a single firewall filter term, you cannot configure both the **loss-priority** action modifier and the **three-color-policer** action modifier. These statements are mutually exclusive.

To enable TCM:

- Enable two-rate tricolor marking.

```
[edit]
user@host# edit class-of-service
user@host# set tri-color
```

Related Documentation

- Overview of Tricolor Marking Architecture on page 171
- Configuring and Applying Tricolor Marking Policers on page 175

Configuring and Applying Tricolor Marking Policers

A tricolor marking (TCM) policer polices traffic on the basis of metering rates, including the CIR, the PIR, their associated burst sizes, and any policing actions configured for the traffic.

This topic describes how to configure and apply TCM policers and includes the following topics:

- Defining a Tricolor Marking Policer on page 175
- Applying Tricolor Marking Policers to Firewall Filters on page 178
- Applying Firewall Filter Tricolor Marking Policers to Interfaces on page 178
- Example: Configuring and Applying a Single-Rate Tricolor Marking Policer on page 179

Defining a Tricolor Marking Policer

To configure a TCM policer, first enable tricolor marking if not already enabled by default (see "Enabling Tricolor Marking and Limitations of Three-Color Policers" on page 173):

You can configure a tricolor policer to discard high loss priority traffic on a logical interface in the ingress or egress direction. statement.

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

Copyright © 2019, Juniper Networks, Inc.

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

- Low
- Medium-high
- High

Table 18 on page 176 describes all the configurable TCM statements.

Table 18: Tricolor Marking Policer Statements

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

Define the TCM policer at the **[edit firewall]** hierarchy level:

1. Create the TCM policer by defining a name for the policer.

```
[edit]
user@host# edit firewall three-color-policer three-color-policer-name
```

2. Discard traffic on a logical interface using tricolor marking policing.

```
[edit firewall three-color-policer name]
user@host# set action loss-priority high then discard
```


Copyright © 2019, Juniper Networks, Inc.

3. Define the filter as a logical interface policer.

```
[edit firewall three-color-policer name]
user@host# set logical-interface-policer
```

4. Configure a single-rate three-color policer in which marking is based on the committed information rate (CIR), committed burst size (CBS), and excess burst size (EBS).

```
[edit firewall three-color-policer name]
user@host# set single-rate (color-aware | color-blind)
user@host# set single-rate committed-information-rate bps
user@host# set single-rate committed-burst-size bytes
user@host# set single-rate excess-burst-size bytes
```

5. Configure a two-rate three-color policer in which marking is based on the committed information rate (CIR), committed burst size (CBS), peak information rate (PIR), and peak burst size (PBS).

```
[edit firewall three-color-policer name]
user@host# set two-rate (color-aware | color-blind)
user@host# set two-rate committed-information-rate bps
user@host# set two-rate committed-burst-size bytes
user@host# set two-rate peak-information-rate bps
user@host# set two-rate peak-burst-size bytes
```

6. Confirm the configuration.

```
[edit firewall]
user@host# show
```

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

Copyright © 2019, Juniper Networks, Inc.

7. Save the configuration.

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

Applying Tricolor Marking Policers to Firewall Filters

To rate-limit traffic by applying a tricolor marking policer to a firewall filter:

- Set the **three-color-policer** statement at the **edit firewall** hierarchy level:

```
[edit]
user@host# edit firewall
user@host# set three-color-policer three-color-policer-name
```

You can include this statement at the following hierarchy levels:

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

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

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

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

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

Applying Firewall Filter Tricolor Marking Policers to Interfaces

To apply a tricolor marking policer to an interface, you must reference the filter name in the interface configuration.

Copyright © 2019, Juniper Networks, Inc.

- Set the **filter** statement:

```
[edit]
user@host# edit interfaces interface-name unit logical-unit-number family family
user@host# set filter input filter-name
user@host# set filter output filter-name
```

NOTE: The filter name that you reference must have an attached tricolor marking policer.

You can include these statements at the following hierarchy levels:

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

Example: Configuring and Applying a Single-Rate Tricolor Marking Policer

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

This example describes how to configure and apply a color-blind, single-rate, tricolor policer.

1. Configure the single-rate, color-blind, three-color policer.

```
[edit]
user@host# edit firewall three-color-policer srtcm1-cb single-rate
user@host# set color-blind
user@host# set committed-information-rate 1048576
user@host# set committed-burst-size 65536
user@host# excess-burst-size 131072
```

2. Apply the policer to the **fil** firewall filter.

```
[edit firewall]
user@host# set filter fil term default then three-color-policer single-rate srtc1-cb
```

3. Apply the **fil** firewall filter to the logical interface:

```
[edit]
user@host# edit interfaces so-1/0/0 unit 0
user@host# set family inet filter input fil
```

4. Verify the configuration.

Copyright © 2019, Juniper Networks, Inc.

```
[edit firewall]
user@host# show
```

```
three-color-policer srtcm1-cb {
    single-rate {
        color-blind;
        committed-information-rate 1048576;
        committed-burst-size 65536;
        excess-burst-size 131072;
    }
}
filter fil {
    term default {
        then {
            three-color-policer {
                single-rate srtcm1-cb;
            }
        }
    }
}
```

```
[edit interfaces]
user@host# show
```

```
so-1/0/0 {
    unit 0 {
        family inet {
            filter {
                input fil;
            }
        }
    }
}
```

5. Save the configuration.

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

Related Documentation

- Controlling Network Access Using Traffic Policing Overview on page 115
- Overview of Tricolor Marking Architecture on page 171

Configuring Single-Rate Tricolor Marking

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

Copyright © 2019, Juniper Networks, Inc.

This topic describes how to configure each mode for single-rate TCM and includes the following sections:

- Configuring Color-Blind Mode for Single-Rate Tricolor Marking on page 181
- Configuring Color-Aware Mode for Single-Rate Tricolor Marking on page 181

Configuring Color-Blind Mode for Single-Rate Tricolor Marking

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

In color-blind mode, packets that exceed the CBS but are below the EBS are marked yellow (medium-high). Packets that exceed the EBS are marked red (high), as shown in Table 19 on page 181.

Table 19: Color-Blind Mode TCM Color-to-PLP Mapping

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

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

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

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

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

Configuring Color-Aware Mode for Single-Rate Tricolor Marking

In color-aware mode, the metering treatment the packet receives depends on its classification. Metering can increase a packet's preassigned PLP, but cannot decrease it, as shown in Table 20 on page 182.

Copyright © 2019, Juniper Networks, Inc.

Table 20: Color-Aware Mode TCM PLP Mapping

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

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

Effect on Low PLP of Single-Rate Policer

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

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

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

Effect on Medium-Low PLP of Single-Rate Policer

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

Copyright © 2019, Juniper Networks, Inc.

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

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

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

Effect on Medium-High PLP of Single-Rate Policer

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

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

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

Effect on High PLP of Single-Rate Policer

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

Related Documentation

- Configuring and Applying Tricolor Marking Policers on page 175
- Configuring Two-Rate Tricolor Marking on page 184

Copyright © 2019, Juniper Networks, Inc.

Configuring Two-Rate Tricolor Marking

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

This topic describes how to configure each mode for two-rate TCM and includes the following sections:

- Configuring Color-Blind Mode for Two-Rate Tricolor Marking on page 184
- Configuring Color-Aware Mode for Two-Rate Tricolor Marking on page 185

Configuring Color-Blind Mode for Two-Rate Tricolor Marking

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

In color-blind mode, packets that exceed the CIR but are below the PIR are marked yellow (medium-high). Packets that exceed the PIR are marked red (high), as shown in Table 21 on page 184.

Table 21: Color-Blind Mode TCM Color-to-PLP Mapping

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

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

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

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

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

Copyright © 2019, Juniper Networks, Inc.

Configuring Color-Aware Mode for Two-Rate Tricolor Marking

In color-aware mode, the metering treatment the packet receives depends on its classification. Metering can increase a packet's preassigned PLP, but cannot decrease it, as shown in Table 22 on page 185.

Table 22: Color-Aware Mode TCM Mapping

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

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

Effect on Low PLP of Two-Rate Policer

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

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

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

Copyright © 2019, Juniper Networks, Inc.

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

Effect on Medium-Low PLP of Two-Rate Policer

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

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

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

Effect on Medium-High PLP of Two-Rate Policer

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

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

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

Effect on High PLP of Two-Rate Policer

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

Related Documentation

- Configuring and Applying Tricolor Marking Policers on page 175
- Configuring Single-Rate Tricolor Marking on page 180

Copyright © 2019, Juniper Networks, Inc.

Example: Configuring and Verifying Two-Rate Tricolor Marking

This topic provides several examples of how you can configure and verify two-rate tricolor marking policers and includes the following sections:

- Requirements on page 187
- Overview on page 187
- Configuration on page 187
- Verification on page 194

Requirements

No special configuration beyond device initialization is required before configuring this example.

Overview

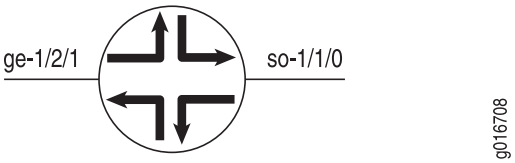
This example configures a two-rate tricolor marking policer on an input Gigabit Ethernet interface and shows commands to verify its operation.

Traffic enters the Gigabit Ethernet interface and exits a SONET/SDH OC12 interface. Oversubscription occurs when you send line-rate traffic from the Gigabit Ethernet interface out the OC12 interface.

Topology

Figure 29 on page 187 shows the sample topology.

Figure 29: Tricolor Marking Sample Topology



Configuration

To configure two-rate tricolor marking policers, perform these tasks:

- Example: Applying a Policer to an Input Interface on page 189
- Example: Applying Profiles to an Output Interface on page 190
- Example: Marking Packets with Medium-Low Loss Priority on page 192
- Results on page 193

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.

Copyright © 2019, Juniper Networks, Inc.

Applying a Policer to an Input Interface

```
set interfaces ge-1/2/1 unit 0 family inet filter input trtcm-filter
set firewall three-color-policer trtcm1 two-rate color-aware
set firewall three-color-policer trtcm1 two-rate committed-information-rate 100m
set firewall three-color-policer trtcm1 two-rate committed-burst-size 65536
set firewall three-color-policer trtcm1 two-rate peak-information-rate 200m
set firewall three-color-policer trtcm1 two-rate peak-burst-size 131072
set firewall filter trtcm-filter term one then three-color-policer two-rate trtcm1
```

Applying Profiles to an Output Interface

```
set class-of-service drop-profiles low-tcm fill-level 80 drop-probability 100
set class-of-service drop-profiles med-tcm fill-level 40 drop-probability 100
set class-of-service drop-profiles high-tcm fill-level 10 drop-probability 100
set class-of-service tri-color
set class-of-service interfaces so-1/1/0 scheduler-map tcm-sched
set class-of-service scheduler-maps tcm-sched forwarding-class queue-0 scheduler
  q0-sched
set class-of-service scheduler-maps tcm-sched forwarding-class queue-3 scheduler
  q3-sched
set class-of-service schedulers q0-sched transmit-rate percent 50
set class-of-service schedulers q0-sched buffer-size percent 50
set class-of-service schedulers q0-sched drop-profile-map loss-priority low protocol any
  drop-profile low-tcm
set class-of-service schedulers q0-sched drop-profile-map loss-priority medium-high
  protocol any drop-profile med-tcm
set class-of-service schedulers q0-sched drop-profile-map loss-priority high protocol any
  drop-profile high-tcm
set class-of-service schedulers q3-sched transmit-rate percent 50
set class-of-service schedulers q3-sched buffer-size percent 50
```

Marking Packets with Medium-Low Loss Priority

```
set interfaces ge-1/2/1 unit 0 family inet filter input 4PLP
set interfaces ge-1/2/1 unit 0 family inet policer input 4PLP
set interfaces ge-1/2/1 unit 0 family inet address 10.45.10.2/30
set firewall three-color-policer trTCM two-rate color-blind
set firewall three-color-policer trTCM two-rate committed-information-rate 400m
set firewall three-color-policer trTCM two-rate committed-burst-size 100m
set firewall three-color-policer trTCM two-rate peak-information-rate 1g
set firewall three-color-policer trTCM two-rate peak-burst-size 500m
set firewall policer 4PLP if-exceeding bandwidth-limit 40k
set firewall policer 4PLP if-exceeding burst-size-limit 4k
set firewall policer 4PLP then loss-priority medium-low
set firewall family inet filter 4PLP term 0 from precedence 1
set firewall family inet filter 4PLP term 0 then loss-priority medium-low
set firewall family inet filter filter_trTCM term default then three-color-policer two-rate
  trTCM
```


Copyright © 2019, Juniper Networks, Inc.

Example: Applying a Policer to an Input Interface

Step-by-Step Procedure

In the following example, the tricolor marking and policer are applied on the ingress Gigabit Ethernet interface. Incoming packets are metered. Packets that do not exceed the CIR are marked with low loss priority. Packets that exceed the CIR, but do not exceed the PIR, are marked with medium-high loss priority. Packets that exceed the PIR are marked with high loss priority.

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

1. Configure the three-color policer.

```
[edit]
user@host# edit firewall three-color-policer trtcm1 two-rate
user@host# set committed-information-rate 100m
user@host# set committed-burst-size 65536
user@host# set peak-information-rate 200m
user@host# set peak-burst-size 131072
```

2. Configure the policer in a firewall filter.

```
[edit]
user@host# set firewall filter trtcm-filter term one then three-color-policer two-rate
   trtcm1
```

3. Apply the firewall filter (policer) as an input filter on the logical interface.

```
[edit]
user@host# edit interfaces ge-1/2/1 unit 0 family inet
user@host# set filter input trtcm-filter
```

4. Confirm the configuration.

```
[edit]
user@host# show
```

```
interfaces {
ge-1/2/1 {
    unit 0 {
        family inet {
            filter {
                input trtcm-filter;
            }
        }
    }
}
firewall {
 three-color-policer trtcm1 {
```

Copyright © 2019, Juniper Networks, Inc.

```
        two-rate {
            color-aware;
            committed-information-rate 100m;
            committed-burst-size 65536;
            peak-information-rate 200m;
            peak-burst-size 131072;
        }
    }

filter trtcm-filter {
        term one {
            then {
                three-color-policer {
                    two-rate trtcm1;
                }
            }
        }
    }
}
```

5. Save the configuration.

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

Example: Applying Profiles to an Output Interface

Step-by-Step Procedure

In the following example, transmission scheduling and weighted random early detection (WRED) profiles are applied on the output OC12 interface. The software drops traffic in the low, medium-high, and high drop priorities proportionally to the configured drop profiles.

1. Define the drop profile.

```
[edit]
user@host# edit class-of-service
user@host# set drop-profiles low-tcm fill-level 80 drop-probability 100
user@host# set drop-profiles med-tcm fill-level 40 drop-probability 100
user@host# set drop-profiles high-tcm fill-level 10 drop-probability 100
user@host# set tri-color
```

2. Specify the scheduler name and parameter values.

```
[edit class-of-service]
user@host# set schedulers q0-sched transmit-rate percent 50
user@host# set schedulers q0-sched buffer-size percent 50
user@host# set schedulers q0-sched drop-profile-map loss-priority low protocol
  any drop-profile low-tcm
user@host# set schedulers q0-sched drop-profile-map loss-priority medium-high
  protocol any drop-profile med-tcm
user@host# set schedulers q0-sched drop-profile-map loss-priority high protocol
  any drop-profile high-tcm
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set schedulers q3-sched transmit-rate percent 50
user@host# set schedulers q3-sched buffer-size percent 50
```

3. Specify a scheduler map name and associate it with the scheduler configuration and forwarding class.

```
[edit class-of-service]
user@host# set scheduler-maps tcm-sched forwarding-class queue-0 scheduler
  q0-sched
user@host# set scheduler-maps tcm-sched forwarding-class queue-3 scheduler
  q3-sched
```

4. Apply the scheduler map to the interface.

```
[edit class-of-service]
user@host# set interfaces so-1/1/0 scheduler-map tcm-sched
```

5. Verify the configuration.

```
[edit class-of-service]
user@host show
```

```
drop-profiles {
    low-tcm {
        fill-level 80 drop-probability 100;
    }
    med-tcm {
        fill-level 40 drop-probability 100;
    }
    high-tcm {
        fill-level 10 drop-probability 100;
    }
}
tri-color;
interfaces {
    so-1/1/0 {
        scheduler-map tcm-sched;
    }
scheduler-maps {
    tcm-sched {
        forwarding-class queue-0 scheduler q0-sched;
        forwarding-class queue-3 scheduler q3-sched;
    }
}
schedulers {
    q0-sched {
        transmit-rate percent 50;
        buffer-size percent 50;
       drop-profile-map loss-priority low protocol any drop-profile low-tcm;

       drop-profile-map loss-priority medium-high protocol any drop-profile
 med-tcm;
       drop-profile-map loss-priority high protocol any drop-profile
high-tcm;
```

Copyright © 2019, Juniper Networks, Inc.

```
    }
    q3-sched {
        transmit-rate percent 50;
        buffer-size percent 50;
    }
}
```

6. Save the configuration.

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

Example: Marking Packets with Medium-Low Loss Priority

Step-by-Step Procedure

In the following example, the 4PLP filter and policer causes certain packets to be marked with medium-low loss priority.

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

1. Configure the firewall filter.

   a. Define the three-color policer.

   ```
   [edit]
   user@host# edit firewall three-color-policer trTCM two-rate
   user@host# set color-blind
   user@host# set committed-information-rate 400m
   user@host# set committed-burst-size 100m
   user@host# set peak-information-rate 1g
   user@host# set peak-burst-size 500m
   ```

   b. Configure policer rate limits and actions.

   ```
   [edit]
   user@host# edit firewall policer 4PLP
   user@host# set if-exceeding bandwidth-limit 40k
   user@host# set if-exceeding burst-size-limit 4k
   user@host# set then loss-priority medium-low
   ```

   c. Configure the IPv4 firewall filter.

   ```
   [edit]
   user@host# edit firewall family inet filter 4PLP term 0
   user@host# set from precedence 1
   user@host# set then loss-priority medium-low
   ```

   d. Define the terms of the IPv4 firewall filter.

   ```
   [edit]
   user@host# edit firewall family inet filter filter_trTCM
   ```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set term default then three-color-policer two-rate trTCM
```

2. Apply the filter to the interface.

```
[edit]
user@host# edit interfaces ge-1/2/1 unit 0 family inet
user@host# set filter input 4PLP
user@host# set policer input 4PLP
user@host# set address 10.45.10.2/30
```

Results

Confirm your configuration by entering the **show interfaces** and **show firewall** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@host# show
```

```
interfaces {
 ge-1/2/1 {
        unit 0 {
            family inet {
                filter {
                    input 4PLP;
                }
                policer {
                    input 4PLP;
                }
                address 10.45.10.2/30;
            }
        }
    }
}
firewall {
  three-color-policer trTCM {
        two-rate {
            color-blind;
            committed-information-rate 400m;
            committed-burst-size 100m;
            peak-information-rate 1g;
            peak-burst-size 500m;
        }
    }
  policer 4PLP {
        if-exceeding {
            bandwidth-limit 40k;
            burst-size-limit 4k;
        }
        then loss-priority medium-low;
    }
  family inet {
    filter 4PLP {
            term 0 {
```

Copyright © 2019, Juniper Networks, Inc.

```
                from {
                    precedence 1;
                }
                then loss-priority medium-low;
            }
        }
    filter trtcm-filter {
        term one {
            then {
                three-color-policer {
                    two-rate trtcm1;
                }
            }
        }
    }
}
```

Verification

Confirm that the configuration is working properly.

- Verifying Two-Rate Tricolor Marking Operation on page 194

Verifying Two-Rate Tricolor Marking Operation

Action The following operational mode commands are useful for checking the results of your configuration:

- **show class-of-service forwarding-table classifiers**
- **show interfaces *interface-name* extensive**
- **show interfaces queue *interface-name***

For information about these commands, see the CLI Explorer.

Related Documentation

- Configuring Two-Rate Tricolor Marking on page 184
- *Guidelines for Configuring Firewall Filters*

Applying Firewall Filter Tricolor Marking Policers to Interfaces

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

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

You can include these statements at the following hierarchy levels:

Copyright © 2019, Juniper Networks, Inc.

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

The filter name that you reference should have an attached tricolor marking policer, as shown in "Configuring and Applying Tricolor Marking Policers" on page 175.

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

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

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

Related Documentation

- Configuring and Applying Tricolor Marking Policers on page 175

Policer Overhead to Account for Rate Shaping in the Traffic Manager

- Policer Overhead to Account for Rate Shaping Overview on page 195
- Example: Configuring Policer Overhead to Account for Rate Shaping on page 196

Policer Overhead to Account for Rate Shaping Overview

If you configure ingress or egress traffic-shaping overhead values for an interface, the traffic manager cannot apply these values to any rate-limiting also applied to the interface. To enable the router to account for the additional Ethernet frame length when policing

Copyright © 2019, Juniper Networks, Inc.

actions are being determined, you must configure the ingress or egress overhead values for policers separately.

NOTE: When a policer overhead value is changed, the PIC or DPC goes offline and then comes back online.

For Gigabit Ethernet Intelligent Queuing 2 (IQ2) and Enhanced IQ2 (IQ2E) PICs or interfaces on Dense Port Concentrators (DPCs) in MX Series routers, you can control the rate of traffic that passes through all interfaces on the PIC or DPC by configuring a *policer overhead*. You can configure a policer ingress overhead and a policer egress overhead, each with values from 0 through 255 bytes. The policer overhead values are added to the length of the final Ethernet frame when determining ingress and egress policer actions.

See Also

- egress-policer-overhead on page 1097
- ingress-policer-overhead on page 1167

Example: Configuring Policer Overhead to Account for Rate Shaping

This example shows how to configure overhead values for policers when rate-shaping overhead is configured.

- Requirements on page 196
- Overview on page 196
- Configuration on page 197
- Verification on page 203

Requirements

Before you begin, make sure that interface for which you are applying ingress or egress policer overhead is hosted on one of the following:

- Gigabit Ethernet IQ2 PIC
- IQ2E PIC
- DPCs in MX Series routers

Overview

This example shows how to configure policer overhead values for all physical interfaces on a supported PIC or MPC so that the rate shaping value configured on a logical interface is accounted for in any policing on that logical interface.

Topology

The router hosts a Gigabit Ethernet IQ2 PIC, installed in PIC location 3 of the Flexible PIC Concentrator (FPC) in slot number 1. The physical interface on port 1 on that PIC is configured to receive traffic on logical interface 0 and send it back out on logical interface 1. Class-of-service scheduling includes 100 Mbps of traffic rate-shaping overhead

Copyright © 2019, Juniper Networks, Inc.

for the output traffic. A policer egress overhead of 100 bytes is configured on the entire PIC so that, for any policers applied to the output traffic, 100 bytes are added to the final Ethernet frame length when determining ingress and egress policer actions.

NOTE:

Traffic rate-shaping and corresponding policer overhead are configured separately:

- You configure rate shaping at the **[edit class-of-service interfaces *interface-name* unit *unit-number*]** hierarchy level.
- You configure policer overhead at the **[edit chassis fpc *slot-number* pic *pic-number*]** hierarchy level.

When a policer overhead value is changed, the PIC or DPC goes offline and then comes back online.

Configuration

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode*.

To configure this example, perform the following tasks:

- Configuring the Logical Interfaces on page 198
- Configuring Traffic Rate-Shaping on the Logical Interface That Carries Output Traffic on page 199
- Configuring Policer Overhead on the PIC or DPC That Hosts the Rate-Shaped Logical Interface on page 201
- Applying a Policer to the Logical Interface That Carries Input Traffic on page 202

CLI Quick Configuration

To quickly configure this example, copy the following configuration commands into a text file, remove any line breaks, and then paste the commands into the CLI at the **[edit]** hierarchy level.

```
set interfaces ge-1/3/1 per-unit-scheduler
set interfaces ge-1/3/1 vlan-tagging
set interfaces ge-1/3/1 unit 0 vlan-id 100
set interfaces ge-1/3/1 unit 0 family inet address 10.10.10.1/30
set interfaces ge-1/3/1 unit 1 vlan-id 101
set interfaces ge-1/3/1 unit 1 family inet address 20.20.20.1/30 arp 20.20.20.2 mac
   00:00:11:22:33:44
set class-of-service schedulers be transmit-rate percent 5
set class-of-service schedulers ef transmit-rate percent 30
set class-of-service schedulers af transmit-rate percent 30
set class-of-service schedulers nc transmit-rate percent 35
set class-of-service scheduler-maps my-map forwarding-class best-effort scheduler be
set class-of-service scheduler-maps my-map forwarding-class expedited-forwarding
   scheduler ef
```

Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service scheduler-maps my-map forwarding-class network-control scheduler
  nc
set class-of-service scheduler-maps my-map forwarding-class assured-forwarding
  scheduler af
set class-of-service interfaces ge-1/3/1 unit 1 scheduler-map my-map
set class-of-service interfaces ge-1/3/1 unit 1 shaping-rate 100m
set firewall policer 500Kbps logical-interface-policer
set firewall policer 500Kbps if-exceeding bandwidth-limit 500k
set firewall policer 500Kbps if-exceeding burst-size-limit 625k
set firewall policer 500Kbps then discard
set chassis fpc 1 pic 3 ingress-policer-overhead 100
set chassis fpc 1 pic 3 egress-policer-overhead 100
set interfaces ge-1/3/1 unit 0 family inet policer input 500Kbps
```

Configuring the Logical Interfaces

Step-by-Step Procedure

To configure the logical interfaces:

1. Enable configuration of the interface

```
[edit]
user@host# edit interfaces ge-1/3/1
```

2. Enable multiple queues for each logical interface (so that you can associate an output scheduler with each logical interface).

```
[edit interfaces ge-1/3/1]
user@host# set per-unit scheduler
user@host# set vlan-tagging
```

NOTE: For Gigabit Ethernet IQ2 PICs only, use the **shared-scheduler** statement to enable shared schedulers and shapers on a physical interface.

3. Configure logical interface **ge-1/3/1.0**.

```
[edit interfaces ge-1/3/1]
user@host# set unit 0 vlan-id 100
user@host# set unit 0 family inet address 10.10.10.1/30
```

4. Configure logical interface **ge-1/3/1.1**.

```
[edit interfaces ge-1/3/1]
user@host# set unit 1 vlan-id 101
user@host# set unit 1 family inet address 20.20.20.1/30 arp 20.20.20.2 mac
  00:00:11:22:33:44
```


Copyright © 2019, Juniper Networks, Inc.

Results Confirm the configuration of the interfaces by entering the **show interfaces** configuration mode command. If the command output does not display the intended configuration, repeat the instructions in this procedure to correct the configuration.

```
[edit]
user@host# show interfaces
ge-1/3/1 {
   per-unit-scheduler;
   vlan-tagging;
   unit 0 {
      vlan-id 100;
      family inet {
         address 10.10.10.1/30;
      }
   }
   unit 1 {
      vlan-id 101;
      family inet {
         address 20.20.20.1/30 {
            arp 20.20.20.2 mac 00:00:11:22:33:44;
         }
      }
   }
}
```

Configuring Traffic Rate-Shaping on the Logical Interface That Carries Output Traffic

Step-by-Step Procedure To configure traffic rate-shaping on the logical interface that carries output traffic:

1. Enable configuration of class-of-service features.

   ```
   [edit]
   user@host# edit class-of-service
   ```

2. Configure packet scheduling on logical interface **ge-1/3/1.0**.

   a. Configure schedulers that specify the percentage of transmission capacity.

      ```
      [edit class-of-service]
      user@host# edit schedulers

      [edit class-of-service schedulers]
      user@host# set be transmit-rate percent 5
      user@host# set ef transmit-rate percent 30
      user@host# set af transmit-rate percent 30
      user@host# set nc transmit-rate percent 35
      ```

      A percentage of zero drops all packets in the queue. When the **rate-limit** option is specified, the transmission rate is limited to the rate-controlled amount. In contrast with the **exact** option, a scheduler with the **rate-limit** option shares unused bandwidth above the rate-controlled amount.

Copyright © 2019, Juniper Networks, Inc.

b. Configure a scheduler map to associate each scheduler with a forwarding class.

```
[edit class-of-service]
user@host# edit scheduler-maps my-map

[edit class-of-service scheduler-maps my-map]
user@host# set forwarding-class best-effort scheduler be
user@host# set forwarding-class expedited-forwarding scheduler ef
user@host# set forwarding-class network-control scheduler nc
user@host# set forwarding-class assured-forwarding scheduler af
```

c. Associate the scheduler map with logical interface **ge-1/3/1.0**.

```
[edit class-of-service]
user@host# edit interfaces ge-1/3/1 unit 1

[edit class-of-service interfaces ge-1/3/1 unit 1]
user@host# set scheduler-map my-map
```

3. Configure 100 Mbps of traffic rate-shaping overhead on logical interface **ge-1/3/1.1**.

```
[edit class-of-service interfaces ge-1/3/1 unit 1]
user@host# set shaping-rate 100
```

Alternatively, you can configure a shaping rate for a logical interface and oversubscribe the physical interface by including the **shaping-rate** statement at the **[edit class-of-service traffic-control-profiles]** hierarchy level. With this configuration approach, you can independently control the delay-buffer rate.

Results Confirm the configuration of the class-of-service features (including the 100 Mbp of shaping of the egress traffic) by entering the **show class-of-service** configuration mode command. If the command output does not display the intended configuration, repeat the instructions in this procedure to correct the configuration.

```
[edit]
user@host# show class-of-service
interfaces {
   ge-1/3/1 {
      unit 1 {
         scheduler-map my-map;
         shaping-rate 100m;
      }
   }
}
scheduler-maps {
   my-map {
      forwarding-class best-effort scheduler be;
      forwarding-class expedited-forwarding scheduler ef;
      forwarding-class network-control scheduler nc;
      forwarding-class assured-forwarding scheduler af;
   }
}
```


Copyright © 2019, Juniper Networks, Inc.

```
schedulers {
  be {
    transmit-rate percent 5;
  }
  ef {
    transmit-rate percent 30;
  }
  af {
    transmit-rate percent 30;
  }
  nc {
    transmit-rate percent 35;
  }
}
```

Configuring Policer Overhead on the PIC or DPC That Hosts the Rate-Shaped Logical Interface

Step-by-Step Procedure

To configure policer overhead on the PIC or MPC that hosts the rate-shaped logical interface:

1. Enable configuration of the supported PIC or MPC.

```
[edit]
user@host# set chassis fpc 1 pic 3
```

2. Configure 100 bytes of policer overhead on the supported PIC or MPC.

```
[edit chassis fpc 1 pic 3]
user@host# set ingress-policer-overhead 100
user@host# set egress-policer-overhead 100
```

NOTE: These values are added to the length of the final Ethernet frame when determining ingress and egress policer actions for all physical interfaces on the PIC or MPC.

You can specify policer overhead with values from 0 through 255 bytes.

Results

Confirm the configuration of the policer overhead on the physical interface to account for rate-shaping by entering the **show chassis** configuration mode command. If the command output does not display the intended configuration, repeat the instructions in this procedure to correct the configuration.

```
[edit]
user@host# show chassis
chassis {
  fpc 1 {
```

Copyright © 2019, Juniper Networks, Inc.

```
      pic 3 {
        egress-policer-overhead 100;
        ingress-policer-overhead 100;
      }
    }
}
```

Applying a Policer to the Logical Interface That Carries Input Traffic

Step-by-Step Procedure

To apply a policer to the logical interface that carries input traffic:

1. Configure the logical interface (aggregate) policer.

```
[edit]
user@host# edit firewall policer 500Kbps

[edit firewall policer 500Kbps]
user@host# set logical-interface-policer
user@host# set if-exceeding bandwidth-limit 500k
user@host# set if-exceeding burst-size-limit 625k
user@host# set then discard
```

2. Apply the policer to Layer 3 input on the IPv4 logical interface.

```
[edit]
user@host# set interfaces ge-1/3/1 unit 0 family inet policer input 500Kbps
```

NOTE: The 100 Mbps policer overhead is added to the length of the final Ethernet frame when determining ingress and egress policer actions,

Results

Confirm the configuration of the policer with rate-shaping overhead by entering the **show firewall** and **show interfaces** configuration mode commands. If the command output does not display the intended configuration, repeat the instructions in this procedure to correct the configuration.

```
[edit]
user@host# show firewall
policer 500Kbps {
  logical-interface-policer;
  if-exceeding {
    bandwidth-limit 500k;
    burst-size-limit 625k;
  }
  then discard;
}
[edit]
user@host# show interfaces
```


Copyright © 2019, Juniper Networks, Inc.

```
ge-1/3/1 {
  per-unit-scheduler;
  vlan-tagging;
  unit 0 {
    vlan-id 100;
    layer2-policer {
      input-policer 500Kbps;
    }
    family inet {
      address 10.10.10.1/30;
    }
  }
  unit 0 {
    vlan-id 101;
    family inet {
      address 20.20.20.1/30 {
        arp 20.20.20.2 mac 00:00:11:22:33:44;
      }
    }
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Displaying Traffic Statistics and Policers for the Logical Interface on page 203
- Displaying Statistics for the Policer on page 204

Displaying Traffic Statistics and Policers for the Logical Interface

Purpose Verify the traffic flow through the logical interface and that the policer is evaluated when packets are received on the logical interface.

Action Use the **show interfaces** operational mode command for logical interface **ge-1/3/1.0**, and include the **detail** or **extensive** option. The command output section for **Traffic statistics** lists the number of bytes and packets received and transmitted on the logical interface, and the **Protocol inet** section contains a **Policer** field that would list the policer **500Kbps** as an input or output policer as follows:

- **Input: 500Kbps-ge-1/3/1.0-log_int-i**
- **Output: 500Kbps-ge-1/3/1.0-log_int-o**

The **log_int-i** suffix denotes a logical interface policer applied to input traffic, while the **log_int-o** suffix denotes a logical interface policer applied to output traffic. In this example, the logical interface policer is applied to Input traffic only.

Copyright © 2019, Juniper Networks, Inc.

Displaying Statistics for the Policer

Purpose Verify the number of packets evaluated by the policer.

Action Use the **show policer** operational mode command and optionally specify the name of the policer. The command output displays the number of packets evaluated by each configured policer (or the specified policer), in each direction. For the policer **500Kbps**, the input and output policer names are displayed as follows:

- **500Kbps-ge-1/3/1.0-log_int-i**
- **500Kbps-ge-1/3/1.0-log_int-o**

The **log_int-i** suffix denotes a logical interface policer applied to input traffic, while the **log_int-o** suffix denotes a logical interface policer applied to output traffic. In this example, the logical interface policer is applied to input traffic only.

See Also
- egress-policer-overhead on page 1097
- ingress-policer-overhead on page 1167

Related Documentation
- *Two-Color Policer Configuration Overview*
- *Guidelines for Applying Traffic Policers*
- "Configuring a Policer Overhead on page 817" in the CLI Explorer

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 5

Defining Forwarding Behavior with Forwarding Classes

- Understanding How Forwarding Classes Assign Classes to Output Queues on page 205
- Default Forwarding Classes on page 208
- Configuring a Custom Forwarding Class for Each Queue on page 212
- Configuring Up to 16 Custom Forwarding Classes on page 214
- Classifying Packets by Egress Interface on page 221
- Forwarding Policy Options Overview on page 223
- Configuring CoS-Based Forwarding on page 224
- Example: Configuring CoS-Based Forwarding on page 227
- Example: Configuring CoS-Based Forwarding for Different Traffic Types on page 230
- Example: Configuring CoS-Based Forwarding for IPv6 on page 230
- Applying Forwarding Classes to Interfaces on page 231
- Understanding Queuing and Marking of Host Outbound Traffic on page 232
- Forwarding Classes and Fabric Priority Queues on page 234
- Default Routing Engine Protocol Queue Assignments on page 234
- Assigning Forwarding Class and DSCP Value for Routing Engine-Generated Traffic on page 237
- Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets on page 238
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 241
- Example: Configuring Different Queuing and Marking Defaults for Outbound Routing Engine and Distributed Protocol Handler Traffic on page 242
- Overriding the Input Classification on page 250

Understanding How Forwarding Classes Assign Classes to Output Queues

This topic covers the following information:

- Output Queue Assignments Based on Forwarding Class on page 206
- Devices That Support Up to Four Forwarding Classes on page 206

Copyright © 2019, Juniper Networks, Inc.

- Devices That Support Up to 16 Forwarding Classes on page 207
- Default and Configurable Packet Loss Priority Values on page 207
- Configuration Statements Used to Configure and Apply Forwarding Classes on page 207

Output Queue Assignments Based on Forwarding Class

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

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

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

Devices That Support Up to Four Forwarding Classes

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

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

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

Copyright © 2019, Juniper Networks, Inc.

Devices That Support Up to 16 Forwarding Classes

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

- EX Series switches
- M7i Multiservices Edge Routers with Enhanced Compact Forwarding Engine Boards (CFEB-Es)
- M10i Multiservices Edge Routers with CFEB-Es
- M120 Multiservices Edge Routers
- M320 Multiservices Edge Routers
- MX Series 5G Universal Routing Platforms
- T Series Core Routers
- PTX Packet Transport Routers

Default and Configurable Packet Loss Priority Values

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

- M320 routers and T Series routers with Enhanced III Flexible PIC Concentrators (FPCs)
- T640 routers with Enhanced Scaling FPC4s
- PTX Series Packet Transport Routers

Configuration Statements Used to Configure and Apply Forwarding Classes

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

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

Copyright © 2019, Juniper Networks, Inc.

```
        forwarding-class class-name;
        forwarding-classes-interface-specific forwarding-class-map-name;
      }
    }
  }
  restricted-queues {
    forwarding-class class-name queue queue-number;
  }
```

Related Documentation

- Default Forwarding Classes on page 208
- Configuring a Custom Forwarding Class for Each Queue on page 212
- Applying Forwarding Classes to Interfaces on page 231
- Configuring Up to 16 Custom Forwarding Classes on page 214
- Controlling Network Access Using Traffic Policing Overview on page 115

Default Forwarding Classes

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

These default mappings apply to all routers. The four forwarding classes defined by default are shown in Table 23 on page 208.

If desired, you can rename the forwarding classes associated with the queues supported on your hardware. Assigning a new class name to an output queue does not alter the default classification or scheduling that is applicable to that queue.

BEST PRACTICE: **CoS configurations can be quite complicated, so unless it is required by your scenario, we recommend that you not alter the default class names or queue number associations.**

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

Table 23: Default Forwarding Classes

Queue	Forwarding Class Name	Comments
Queue 0	**best-effort** (**be**)	The software does not apply any special CoS handling to packets with 000000 in the DiffServ field, a backward compatibility feature. These packets are usually dropped under congested network conditions.

Copyright © 2019, Juniper Networks, Inc.

Table 23: Default Forwarding Classes (continued)

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

The following rules govern queue assignment:

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

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

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

Copyright © 2019, Juniper Networks, Inc.

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

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

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

On Juniper Networks M320 Multiservice Edge Routers, MX Series 5G Universal Routing Platforms, and T Series Core Routers, you can assign multiple forwarding classes to a single queue. If you do so, the first forwarding class that you assign to queue 0 acquires the default BE classification and scheduling. The first forwarding class that you assign to queue 1 acquires the default EF classification and scheduling. The first forwarding class that you assign to queue 2 acquires the default AF classification and scheduling. The first forwarding class that you assign to queue 3 acquires the default NC classification and scheduling. For more information, see "Configuring Up to 16 Custom Forwarding Classes" on page 214.

CAUTION: When you define a forwarding class for the same queue as one of the default forwarding classes, the default forwarding class is automatically removed. For example, if you define class be for queue 0, which is the queue for the default best-effort forwarding class, the best-effort class is removed.

If you define more than one forwarding class for a given queue number and use the name of a default forwarding class for one of the new classes, the new class with the default name is deleted.

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

Copyright © 2019, Juniper Networks, Inc.

you configure traffic through logical unit 1 to be classified into queue 3. You want traffic through logical unit 2 to be classified into the default classifier, which is queue 0. In this case, traffic through logical unit 2 is classified into queue 3, because the configuration of logical unit 1 was committed last.

Related Documentation

- Understanding How Forwarding Classes Assign Classes to Output Queues on page 205
- Configuring a Custom Forwarding Class for Each Queue on page 212
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 241
- CoS Features and Limitations on M Series and T Series Routers on page 553
- CoS Features and Limitations on PTX Series Routers on page 599

Copyright © 2019, Juniper Networks, Inc.

Configuring a Custom Forwarding Class for Each Queue

Copyright © 2019, Juniper Networks, Inc.

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

BEST PRACTICE: CoS configurations can be quite complicated, so unless it is required by your scenario, we recommend that you not alter the default class names or queue number associations.

If your network requires more than the four default forwarding classes, you can use the following procedure to create custom forwarding class names and assign each forwarding class to any queue number by including the **forwarding-classes** statement at the **[edit class-of-service]** hierarchy level.

The **class** and **queue** statements at the **[edit class-of-service forwarding-classes]** hierarchy level are mutually exclusive. If you want to configure one-to-one mapping of forwarding classes to output queues for up to eight forwarding classes, use the **queue** statement at the **[edit class-of-service forwarding-classes]** hierarchy level. If you want to configure up to 16 forwarding classes with multiple forwarding classes mapped to single output queues (see "Configuring Up to 16 Custom Forwarding Classes" on page 214), include the **class** statement at the **[edit class-of-service forwarding-classes]** hierarchy level.

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

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

A default congestion and transmission control mechanism is used when an output interface is not configured for a certain forwarding class, but receives packets destined for that unconfigured forwarding class. This default mechanism uses the delay buffer and weighted round robin (WRR) credit allocated to the designated forwarding class, with a default drop profile. Because the buffer and WRR credit allocation is minimal, packets might be lost if a larger number of packets are forwarded without configuring the forwarding class for the interface.

CAUTION: When you define a forwarding class for the same queue as one of the default forwarding classes, the default forwarding class is automatically removed. For example, if you define class be for queue 0, which is the queue for the default best-effort forwarding class, the best-effort class is removed.

Copyright © 2019, Juniper Networks, Inc.

If you define more than one forwarding class for a given queue number and use the name of a default forwarding class for one of the new classes, the new class with the default name is deleted.

To create custom forwarding class names and assign each forwarding class to any queue number:

1. Access the CoS forwarding class configuration hierarchy.

```
[edit]
user@host# edit class-of-service forwarding-classes
```

2. Specify the queue number and forwarding class name.

```
[edit class-of-service forwarding-classes]
user@host# set queue queue-num class-name
```

Related Documentation

- Understanding How Forwarding Classes Assign Classes to Output Queues on page 205
- Default Forwarding Classes on page 208
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 241

Configuring Up to 16 Custom Forwarding Classes

By default on all routers, four forwarding classes are mapped to four output queues, as shown in the topic "Default Forwarding Classes" on page 208. On M120 and M320 Multiservice Edge Routers, MX Series 5G Universal Routing Platforms, PTX Series Packet Transport Routers, and T Series Core Routers, you can configure more than four forwarding classes and queues; you can configure up to 16 forwarding classes and eight queues, with multiple forwarding classes assigned to single queues. The concept of assigning multiple forwarding classes to a queue is sometimes referred to as creating *forwarding-class aliases*.

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

Mapping multiple forwarding classes to single queues is useful. Suppose, for example, that forwarding classes are set based on multifield packet classification, and the multifield classifiers are different for core-facing interfaces and customer-facing interfaces. Suppose you need four queues for a core-facing interface and five queues for a customer-facing interface, where **fc0** through **fc4** correspond to the classifiers for the customer-facing interface, and **fc5** through **fc8** correspond to classifiers for the core-facing interface, as shown in Figure 30 on page 215.

Copyright © 2019, Juniper Networks, Inc.

Figure 30: Customer-Facing and Core-Facing Forwarding Classes

In this example, you need nine classifiers and, therefore, nine forwarding classes. The forwarding class-to-queue mapping is shown in Table 24 on page 215.

Table 24: Sample Forwarding Class-to-Queue Mapping

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

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

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

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

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

Copyright © 2019, Juniper Networks, Inc.

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

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

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

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

- When you use CoS-based forwarding features, you cannot configure more than eight forwarding classes with a forwarding policy. However, if you try to configure CoS-based forwarding with more than eight forwarding classes configured, commit fails with a message. Therefore, you can configure CBF on a router with eight or less than eight forwarding classes only. Under this condition, the forwarding class to queue mapping can be either one-to-one or one-to-many.
- A scheduler map that maps eight different forwarding classes to eight different schedulers can only be applied to interfaces that support eight queues. If you apply this type of scheduler map to an interface that only supports four queues, then the commit fails.
- We recommend that you configure the statements changing PICs to support eight queues and then applying an eight queue scheduler map in two separate steps. Otherwise, the commit might succeed but the PIC might not have eight queues when the scheduler map is applied, generating an error.

You can determine the ID number assigned to a forwarding class by issuing the **show class-of-service forwarding-class** command. You can determine whether the classification is fixed by issuing the **show class-of-service forwarding-table classifier mapping** command. In the command output, if the **Table Type** field appears as **Fixed**, the classification is fixed. For more information about fixed classification, see "Applying Forwarding Classes to Interfaces" on page 231.

For information about configuring eight forwarding classes on ATM2 IQ interfaces, see "Enabling Eight Queues on ATM Interfaces" on page 842.

- Enabling Eight Queues on Interfaces on page 217
- Assigning Multiple Forwarding Classes and Default Forwarding Classes on page 218
- Examples: Configuring Up to 16 Forwarding Classes on page 219

Copyright © 2019, Juniper Networks, Inc.

Enabling Eight Queues on Interfaces

By default, Intelligent Queuing (IQ), Intelligent Queuing 2 (IQ2), Intelligent Queuing Enhanced (IQE), and Intelligent Queuing 2 Enhanced (IQ2E) PICs on M320 and T Series routers are restricted to a maximum of four egress queues per interface. The following procedures describe how to configure a maximum of eight egress queues on these interfaces.

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

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

To configure a maximum of eight egress queues on these PICs:

1. Specify the PIC you want to configure.

```
[edit]
user@host# edit chassis fpc slot-number pic pic-number
```

2. Configure a maximum of eight egress queues on these interfaces.

```
[edit chassis fpc slot-number pic pic-number]
user@host# set max-queues-per-interface 8
```

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

This procedure describes how to configure the maximum number of queues the interface supports on a TX Matrix or TX Matrix Plus router.

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

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

1. To configure a maximum of eight egress queues on these PICs:

```
[edit]
user@host# edit chassis lcc number fpc slot-number pic pic-number
```

Copyright © 2019, Juniper Networks, Inc.

2. Configure a maximum of eight egress queues on these interfaces.

```
[edit chassis fpc slot-number pic pic-number]
user@host# set max-queues-per-interface 8
```

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

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

1. To view how many queues an interface supports:

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

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

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

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

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

Assigning Multiple Forwarding Classes and Default Forwarding Classes

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

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

Copyright © 2019, Juniper Networks, Inc.

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

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

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

For more information about forwarding class defaults, see "Default Forwarding Classes" on page 208.

Examples: Configuring Up to 16 Forwarding Classes

To configure 16 forwarding classes, map two forwarding classes to each queue. For example:

1. Specify each forwarding class and queue you want mapped.

```
[edit]
user@host# edit class-of-service forwarding-classes
user@host# set class fc0 queue-num 0
user@host# set class fc1 queue-num 0
user@host# set class fc2 queue-num 1
user@host# set class fc3 queue-num 1
user@host# set class fc4 queue-num 2
user@host# set class fc5 queue-num 2
user@host# set class fc6 queue-num 3
user@host# set class fc7 queue-num 3
user@host# set class fc8 queue-num 4
user@host# set class fc9 queue-num 4
user@host# set class fc10 queue-num 5
user@host# set class fc11 queue-num 5
user@host# set class fc12 queue-num 6
user@host# set class fc13 queue-num 6
user@host# set class fc14 queue-num 7
user@host# set class fc15 queue-num 7
```

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

1. Specify each forwarding class and queue you want mapped.

```
[edit]
```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# edit class-of-service restricted-queues
user@host# set forwarding-class fc0 queue 0
user@host# set forwarding-class fc1 queue 0
user@host# set forwarding-class fc2 queue 0
user@host# set forwarding-class fc3 queue 0
user@host# set forwarding-class fc4 queue 1
user@host# set forwarding-class fc5 queue 1
user@host# set forwarding-class fc6 queue 1
user@host# set forwarding-class fc7 queue 1
user@host# set forwarding-class fc8 queue 2
user@host# set forwarding-class fc9 queue 2
user@host# set forwarding-class fc10 queue 2
user@host# set forwarding-class fc11 queue 2
user@host# set forwarding-class fc12 queue 3
user@host# set forwarding-class fc13 queue 3
user@host# set forwarding-class fc14 queue 3
user@host# set forwarding-class fc15 queue 3
```

If you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler. To configure a scheduler map applicable to an interface restricted to four queues:

1. Specify a scheduler map name and associate it with the scheduler configuration and forwarding class.

```
[edit]
user@host# edit class-of-service scheduler-maps interface-restricted
user@host# set forwarding-class be scheduler 00
user@host# set forwarding-class ef scheduler Q1
user@host# set forwarding-class ef1 scheduler Q1
user@host# set forwarding-class ef2 scheduler Q1
user@host# set forwarding-class af1 scheduler Q2
user@host# set forwarding-class af scheduler Q2
user@host# set forwarding-class nc scheduler Q3
user@host# set forwarding-class nc1 scheduler Q3
```

2. Map the forwarding classes to the restricted queues.

```
[edit]
user@host# edit class-of-service restricted-queues
user@host# set forwarding-class be queue 0
user@host# set forwarding-class ef queue 1
user@host# set forwarding-class ef1 queue 1
user@host# set forwarding-class ef2 queue 1
user@host# set forwarding-class af queue 2
user@host# set forwarding-class af1 queue 2
user@host# set forwarding-class nc queue 3
user@host# set forwarding-class nc1 queue 3
```

Related Documentation

- Understanding How Forwarding Classes Assign Classes to Output Queues on page 205

Copyright © 2019, Juniper Networks, Inc.

- Default Forwarding Classes on page 208

Classifying Packets by Egress Interface

For Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers with the Intelligent Queuing (IQ), IQ2, Enhanced IQ (IQE), Multiservices link services intelligent queuing (LSQ) interfaces, or ATM2 PICs, you can classify unicast and multicast packets based on the egress interface. For unicast traffic, you can also use a multifield filter, but only egress interface classification applies to multicast traffic as well as unicast traffic. If you configure egress classification of an interface, you cannot perform Differentiated Services code point (DSCP) rewrites on the interface. By default, the system does not perform any classification based on the egress interface.

On an MX Series router that contains MPCs and MS-DPCs, multicast packets are dropped on the router and not processed properly if the router contains MLPPP LSQ logical interfaces that function as multicast receivers and if the network services mode is configured as enhanced IP mode on the router. This behavior is expected with LSQ interfaces in conjunction with enhanced IP mode. In such a scenario, if enhanced IP mode is not configured, multicasting works correctly. However, if the router contains redundant LSQ interfaces and enhanced IP network services mode configured with FIB localization, multicast works properly.

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

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

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

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

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

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

Copyright © 2019, Juniper Networks, Inc.

All parameters relating to the queues and forwarding class must be configured as well. For more information about configuring forwarding classes and queues, see "Configuring a Custom Forwarding Class for Each Queue" on page 212.

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

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

[edit class-of-service]
interfaces {
   ge-6/0/0 unit 0 {
      output-forwarding-class-map FCMAP1;
   }
}
```

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

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

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

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

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

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

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


Copyright © 2019, Juniper Networks, Inc.

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

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

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

Forwarding Policy Options Overview

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

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

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

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

Copyright © 2019, Juniper Networks, Inc.

```
}
```

NOTE: Beginning with Junos OS Release 17.1R1, QFX10000 Series switches support CoS-based forwarding. **[set class-of-service forwarding-policy class]** is not supported on QFX10000 Series switches.

Beginning with Junos OS Release 17.2, MX routers with MPCs or MS-DPCs, VMX, PTX3000 routers, PTX5000 routers, and VPTX support configuring CoS-based forwarding (CBF) for up to 16 forwarding classes. All other platforms support CBF for up to 8 forwarding classes. To support up to 16 forwarding classes for CBF on MX routers, enable **enhanced-ip** at the **[edit chassis network-services]** hierarchy level. Enabling **enhanced-ip** is not necessary on PTX routers to support 16 forwarding classes for CBF.

Release History Table

Release	Description
17.2R1	Beginning with Junos OS Release 17.2, MX routers with MPCs or MS-DPCs, VMX, PTX3000 routers, PTX5000 routers, and VPTX support configuring CoS-based forwarding (CBF) for up to 16 forwarding classes.
17.1R1	Beginning with Junos OS Release 17.1R1, QFX10000 Series switches support CoS-based forwarding. **[set class-of-service forwarding-policy class]** is not supported on QFX10000 Series switches.

Related Documentation

- Configuring CoS-Based Forwarding on page 224
- Example: Configuring CoS-Based Forwarding on page 227

Configuring CoS-Based Forwarding

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

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

This configuration specifies that routes matching the route filter are subject to the CoS next-hop mapping specified by ***map-name***. For more information about configuring policy statements, see the *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide*.

Copyright © 2019, Juniper Networks, Inc.

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

Beginning with Junos OS Release 17.2, MX routers with MPCs or MS-DPCs, VMX, PTX3000 routers, and PTX5000 routers support configuring CoS-based forwarding (CBF) for up to 16 forwarding classes. All other platforms support CBF for up to 8 forwarding classes. To support up to 16 forwarding classes for CBF on MX routers, enable **enhanced-ip** at the **[edit chassis network-services]** hierarchy level.

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

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

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

When you configure CBF with OSPF as the interior gateway protocol (IGP), you must specify the next hop as an interface name or next-hop alias, not as an IPv4 or IPv6 address. This is true because OSPF adds routes with the interface as the next hop for point-to-point interfaces; the next hop does not contain the IP address. For an example configuration, see "Example: Configuring CoS-Based Forwarding" on page 227.

For Layer 3 VPNs, when you use class-based forwarding for the routes received from the far-end provider edge (PE) router within a VRF instance, the software can match the routes based on the attributes that come with the received route only. In other words, the matching can be based on the route within RIB-in. In this case, the **route-filter**

Copyright © 2019, Juniper Networks, Inc.

statement you include at the **[edit policy-options policy-statement my-cos-forwarding from]** hierarchy level has no effect because the policy checks the **bgp.l3vpn.0** table, not the ***vrf*.inet.0** table.

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

```
[edit class-of-service forwarding-policy]
next-hop-map map1 {
  forwarding-class expedited-forwarding {
    next-hop next-hop1;
    next-hop next-hop2;
  }
  forwarding-class best-effort {
    next-hop next-hop3;
    lsp-next-hop lsp-next-hop4;
  }
  forwarding-class-default {
    lsp-next-hop lsp-next-hop5;
  }
}
```

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

This configuration has the following implications:

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

Copyright © 2019, Juniper Networks, Inc.

lsp, **lsp1**, **lsp2**, and **lsp3**. If you do not want this behavior, you must use the anchor characters **lsp-next-hop " ^lsp$"**, which match **lsp** only.

- The route filter does not work because the policy checks against the **bgp.l3vpn.0** table instead of the ***vrf*.inet.0** table.

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

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

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

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

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

Release History Table

Release	Description
17.2R1	Beginning with Junos OS Release 17.2, MX routers with MPCs or MS-DPCs, VMX, PTX3000 routers, and PTX5000 routers support configuring CoS-based forwarding (CBF) for up to 16 forwarding classes.

Related Documentation

- *Load Balancing VPLS Non-Unicast Traffic Across Member Links of an Aggregate Interface*
- Forwarding Policy Options Overview on page 223

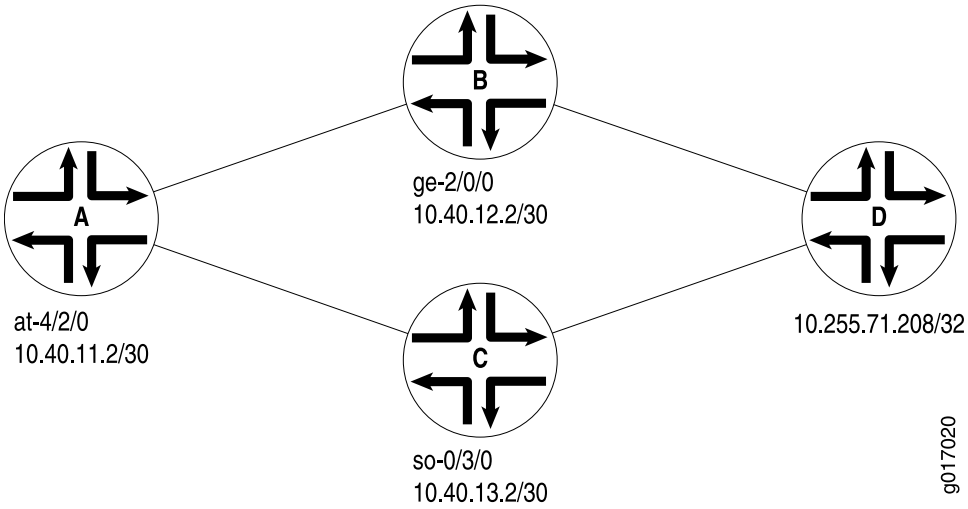
Example: Configuring CoS-Based Forwarding

Router A has two routes to destination **10.255.71.208** on Router D. One route goes through Router B, and the other goes through Router C, as shown in Figure 31 on page 228.

Configure Router A with CoS-based forwarding (CBF) to select Router B for queue 0 and queue 2, and Router C for queue 1 and queue 3.

Copyright © 2019, Juniper Networks, Inc.

Figure 31: Sample CoS-Based Forwarding



When you configure CBF with OSPF as the IGP, you must specify the next hop as an interface name, not as an IPv4 or IPv6 address. The next hops in this example are specified as **ge-2/0/0.0** and **so-0/3/0.0**.

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


Copyright © 2019, Juniper Networks, Inc.

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

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

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

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

Copyright © 2019, Juniper Networks, Inc.

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

Related Documentation

- Forwarding Policy Options Overview on page 223

Example: Configuring CoS-Based Forwarding for Different Traffic Types

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

Only the forwarding policy is shown in this example:

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

Example: Configuring CoS-Based Forwarding for IPv6

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

You can configure a next-hop map with both IPv4 and IPv6 addresses, or you can configure separate next-hop maps for IPv4 and IPv6 addresses and include the **from family (inet**

Copyright © 2019, Juniper Networks, Inc.

| inet6) statements at the **[edit policy-options policy-options policy-statement** ***policy-name*** **term** ***term-name*****]** hierarchy level to ensure that only next-hop maps of a specified protocol are applied to a specified route.

If you do not configure separate next-hop maps and include the **from family (inet | inet6)** statements in the configuration, when a route uses two next hops (whether IPv4, IPv6, interface, or LSP next hop) in at least two of the specified forwarding classes, CBF is used for the route; otherwise, the CBF policy is ignored.

1. Define the CBF next-hop map:

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

2. Define the CBF forwarding policy:

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

3. Export the CBF forwarding policy:

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

Applying Forwarding Classes to Interfaces

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

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

Copyright © 2019, Juniper Networks, Inc.

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

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

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

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

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

Related Documentation

- forwarding-class on page 1134

Understanding Queuing and Marking of Host Outbound Traffic

This topic covers the following information:

- Host Outbound Traffic Overview on page 232
- Default Queuing and Marking of Host Outbound Traffic on page 233
- Configured Queuing and Marking of Host Outbound Traffic on page 233
- Configured Queuing and Marking of Outbound Routing Engine Traffic Only on page 233

Host Outbound Traffic Overview

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

Routing Engine Sourced Traffic

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

Distributed Protocol Handler Traffic

Distributed protocol handler traffic refers to traffic from the router's *periodic packet management* (PPM) process when it runs sessions distributed to the Packet Forwarding Engine (the default mode) in addition to the Routing Engine. The PPM process is responsible for periodic transmission of protocol Hello or other keepalive packets on behalf of its various client processes, such as Bidirectional Forwarding Detection (BFD)

Copyright © 2019, Juniper Networks, Inc.

Protocol or Link Aggregation Control Protocol (LACP), and it also receives packets on behalf of client processes. In addition, PPM handles time-sensitive periodic processing and performs such tasks as sending process-specific packets and gathering statistics. By default, PPM sessions on the Routing Engine run distributed on the Packet Forwarding Engine, and this enables client processes to run on the Packet Forwarding Engine.

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

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

Default Queuing and Marking of Host Outbound Traffic

By default, the router assigns host outbound traffic to the **best-effort** forwarding class (which maps to queue 0) or to the **network-control** forwarding class (which maps to queue 3) based on protocol. For more information, see "Default Routing Engine Protocol Queue Assignments" on page 234.

By default, the router marks the type of service (ToS) field of Layer 3 packets in the host outbound traffic flow with DifffServ code point (DSCP) bits 000000 (which correlate with the **best-effort** forwarding class). The router does not remark Layer 2 traffic such as LACP control traffic on aggregated Ethernet. For more information, see "Default DSCP and DSCP IPv6 Classifiers" on page 41.

Configured Queuing and Marking of Host Outbound Traffic

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

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

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

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

Configured Queuing and Marking of Outbound Routing Engine Traffic Only

To configure a nondefault forwarding class and DSCP bits that the router uses to queue and remark traffic generated by the Routing Engine only, attach an IPv4 firewall filter to

Copyright © 2019, Juniper Networks, Inc.

the output of the router's loopback address. Use the **forwarding-class** and **dscp** filter actions to specify override values.

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

Related Documentation

- Default Routing Engine Protocol Queue Assignments on page 234
- Default DSCP and DSCP IPv6 Classifiers on page 41
- Example: Configuring Different Queuing and Marking Defaults for Outbound Routing Engine and Distributed Protocol Handler Traffic on page 242

Forwarding Classes and Fabric Priority Queues

This topic covers the following information:

- Default Fabric Priority Queuing on page 234
- Overriding Default Fabric Priority Queuing on page 234

Default Fabric Priority Queuing

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

Overriding Default Fabric Priority Queuing

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

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

Related Documentation

- Associating Schedulers with Fabric Priorities on page 331

Default Routing Engine Protocol Queue Assignments

Table 25 on page 235 lists the default output queues to which Routing Engine sourced traffic is mapped by protocol type. In general, control protocol packets are sent over queue 3 and management traffic is sent over queue 0. The following caveats apply to these default queue assignments:

- For all packets sent to queue 3 over a VLAN-tagged interface, the software sets the 802.1p bit to 110, except for VRRP packets, in which case the software sets the 802.1p bit to 111.

Copyright © 2019, Juniper Networks, Inc.

- For IPv4 and IPv6 packets, the software copies the IP type-of-service (ToS) value into the 802.1p field independently of which queue the packets are sent out.
- For MPLS packets, the software copies the EXP bits into the 802.1p field.

Table 25: Default Queue Assignments for Packets Generated by the Routing Engine

Routing Engine Protocol	Default Queue Assignment
Adaptive Services PIC TCP tickle (keepalive packets for idle session generated with stateful firewall to probe idle TCP sessions)	Queue 0
Address Resolution Protocol (ARP)	Queue 0
ATM Operation, Administration, and Maintenance (OAM)	Queue 3
Bidirectional Forwarding Detection (BFD) Protocol	Queue 3
BGP	Queue 0
BGP TCP Retransmission	Queue 3
Cisco High-Level Data Link Control (HDLC)	Queue 3
Distance Vector Multicast Routing Protocol (DVMRP)	Queue 3
Ethernet Operation, Administration, and Maintenance (OAM)	Queue 3
Frame Relay Local Management Interface (LMI)	Queue 3
Frame Relay Asynchronization permanent virtual circuit (PVC)/data link connection identifier (DLCI) status messages	Queue 3
FTP	Queue 0
IS-IS Open Systems Interconnection (OSI)	Queue 3
Internet Control Message Protocol (ICMP)	Queue 0
Internet Group Management Protocol (IGMP) query	Queue 3
IGMP Report	Queue 0
Internet Key Exchange (IKE)	Queue 3
IP version 6 (IPv6) Neighbor Solicitation	Queue 3
IPv6 Neighbor Advertisement	Queue 3
IPv6 Router Advertisement	Queue 0

Copyright © 2019, Juniper Networks, Inc.

Table 25: Default Queue Assignments for Packets Generated by the Routing Engine (continued)

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

Copyright © 2019, Juniper Networks, Inc.

Table 25: Default Queue Assignments for Packets Generated by the Routing Engine (continued)

Routing Engine Protocol	Default Queue Assignment
Virtual Router Redundancy Protocol (VRRP)	Queue 3
xnm-clear-text	Queue 0
xnm-ssl	Queue 0

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

You can set the forwarding class and differentiated service code point (DSCP) value for traffic originating in the Routing Engine. To configure forwarding class and DSCP values that apply to Routing Engine–generated traffic only, apply an output filter to the loopback (**lo.0**) interface and set the appropriate forwarding class and DSCP bit configuration for various protocols. For example, you can set the DSCP value on OSPF packets that originate in the Routing Engine to **10** and assign them to the AF (assured forwarding) forwarding class while the DSCP value on ping packets are set to **0** and use forwarding class BE (best effort).

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

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

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

[edit firewall filter family inet]
filter loopback-filter {
   term t1 {
      from {
         protocol icmp; # For pings
      }
      then {
         forwarding-class af17;
         dscp 38;
      }
   }
```

Copyright © 2019, Juniper Networks, Inc.

```
  term t2 {
    from {
      protocol ospf; # For OSPF
    }
    then {
      forwarding-class af11;
      dscp 12;
    }
  }
  term t3 {
    from {
      protocol tcp; # For BGP
    }
    then {
      forwarding-class af16;
      dscp 10;
    }
  }
  term t4 {
    then accept; # Do not forget!
  }
}

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

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

Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets

On Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers, you can selectively set the DSCP field of MPLS-tagged IPv4 and IPv6 packets to **000000**. In the same packets, you can set the MPLS EXP field according to a configured rewrite table, which is based on the forwarding classes that you set in incoming packets using a BA or multifield classifier.

Queue selection is based on the forwarding classes you assign in scheduler maps. This means that you can direct traffic to a single output queue, regardless of whether the DSCP field is unchanged or rewritten to **000000**. To do this, you must configure a

Copyright © 2019, Juniper Networks, Inc.

multifield classifier that matches selected packets and modifies them with the **dscp 0** action.

Selective marking of DSCP fields to **0**, without affecting output queue assignment, can be useful. For example, suppose you need to use the MPLS EXP value to configure CoS applications for core provider routers. At the penultimate egress provider edge (PE) router where the MPLS labels are removed, the CoS bits need to be provided by another value, such as DSCP code points. This case illustrates why it is useful to mark both the DSCP and MPLS EXP fields in the packet. Furthermore, it is useful to be able to mark the two fields differently, because the CoS rules of the core provider router might differ from the CoS rules of the egress penultimate router. At egress, as always, you can use a rewrite table to rewrite the MPLS EXP values corresponding to the forwarding classes that you need to set.

NOTE: When both customer-facing and core-facing interfaces exist, you can derive the EXP value in the following precedence order, while adding the MPLS label:

1. EXP value provided by the CoS rewrite action.
2. EXP value derived from the top label of the stack (MPLS label stacking).
3. IPv4 or IPv6 precedence (Layer 3 VPN, Layer 2 VPN, and VPLS scenarios).

For IPv4 traffic, the **dscp 0** action modifier at the **[edit firewall family inet filter *filter-name* term *term-name* then]** hierarchy level is valid. However, for IPv6 traffic, you configure this feature by including the **traffic-class 0** action modifier at the **[edit firewall family inet6 filter *filter-name* term *term-name* then]** hierarchy level.

In the following IPv4 example, term **1** of the multifield classifier matches packets with DSCP **001100** code points coming from a certain VRF, rewrites the bits to DSCP **000000**, and sets the forwarding class to **best-effort**. In term **2**, the classifier matches packets with DSCP **010110** code points and sets the forwarding class to **best-effort**. Because term **2** does not include the **dscp 0** action modifier, the DSCP **010110** bits remain unchanged. Because the classifier sets the forwarding class for both code points to **best-effort**, both traffic types are directed to the same output queue.

NOTE: If you configure a bit string in a DSCP match condition in a firewall filter, then you must include the letter “b” in front of the string, or the match rule creation fails on commit.

```
[edit]
firewall {
   family inet {
      filter vrf-rewrite {
```

Copyright © 2019, Juniper Networks, Inc.

```
        term 1 {
          from {
            dscp b001100;
          }
          then {
            dscp 0;
            forwarding-class best-effort;
          }
        }
        term 2 {
          from {
            dscp b010110;
          }
          then {
            forwarding-class best-effort;
          }
        }
      }
    }
  }
```

Applying the Multifield Classifier

Apply the filter to an input interface corresponding to the VRF:

```
[edit]
interfaces {
  so-0/1/0 {
    unit 0 {
      family inet {
        filter input vrf-rewrite;
      }
    }
  }
}
```


Copyright © 2019, Juniper Networks, Inc.

NOTE: The **dscp 0** action is supported in both input and output filters. You can use this action for non-MPLS packets as well as for IPv4 and IPv6 packets entering an MPLS network. All IPv4 and IPv6 firewall filter match conditions are supported with the **dscp 0** action.

The following limitations apply:

- You can use a multifield classifier to rewrite DSCP fields to value 0 only. Other values are not supported.
- If a packet matches a filter that has the **dscp 0** action, then the outgoing DSCP value of the packet is 0, even if the packet matches a rewrite rule, and the rewrite rule is configured to mark the packet to a non-zero value. The **dscp 0** action overrides any other rewrite rule actions configured on the router.
- Although you can use the **dscp 0** action on an input filter, the output filter and other classifiers do not see the packet as being marked **dscp 0**. Instead, they classify the packet based on its original incoming DSCP value. The DSCP value of the packet is set to 0 after all other classification actions have completed on the packet.

Changing the Default Queuing and Marking of Host Outbound Traffic

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

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

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

The queue selected must be properly configured on all interfaces.

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

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

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

Copyright © 2019, Juniper Networks, Inc.

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

Related Documentation

- Understanding How Forwarding Classes Assign Classes to Output Queues on page 205
- Default Routing Engine Protocol Queue Assignments on page 234
- Default DSCP and DSCP IPv6 Classifiers on page 41

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

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

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

- Requirements on page 242
- Overview on page 242
- Configuration on page 243
- Verification on page 247

Requirements

This example uses the following hardware and software components:

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

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

Overview

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

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

Copyright © 2019, Juniper Networks, Inc.

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

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

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

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

Configuration

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

- Configuring R1 Packet Counting on page 244
- Configuring R2 Queuing and Re-Marking of Host Outbound Traffic on page 244
- Configuring R2 Queuing and Re-Marking of Routing Engine Sourced Traffic on page 245

CLI Quick Configuration

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

Router R1

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

Router R2

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

Copyright © 2019, Juniper Networks, Inc.

Configuring R1 Packet Counting

Step-by-Step Procedure

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

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

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

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

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

3. Apply the firewall filter to all egress packets.

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

Configuring R2 Queuing and Re-Marking of Host Outbound Traffic

Step-by-Step Procedure

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

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

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

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

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

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

```
[edit]
```


Copyright © 2019, Juniper Networks, Inc.

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

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

Step-by-Step Procedure

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

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

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

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

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

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

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

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

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

Results

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

Router R1

```
user@R1# show firewall
family inet {
   filter f_bfd_source {
      term 1 {
         from {
            forwarding-class control-traffic;
         }
```

Copyright © 2019, Juniper Networks, Inc.

```
        then {
          count c_sent_bfd;
          accept;
        }
      }
      term 2 {
        from {
          forwarding-class-except control-traffic;
        }
        then {
          count c_sent_other;
          accept;
        }
      }
    }
  }
```

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

Router R2

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

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

```
user@R2# show interfaces
lo0 {
  unit 0 {
    family inet {
```


Copyright © 2019, Juniper Networks, Inc.

```
        filter {
          output f_out_loopback;
        }
      }
    }
}
```

Verification

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

Confirm that the configuration is working properly.

- Verifying the Queue Assignment of the Traffic That R1 Is Sending in the BFD Session on page 247
- Verifying That Router R1 Is Sending BFD Traffic on page 248
- Verifying That Router R2 Is Receiving BFD Traffic on page 249

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

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

Copyright © 2019, Juniper Networks, Inc.

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

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

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

Verifying That Router R1 Is Sending BFD Traffic

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

Copyright © 2019, Juniper Networks, Inc.

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

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

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

Verifying That Router R2 Is Receiving BFD Traffic

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

Copyright © 2019, Juniper Networks, Inc.

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

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

03:23:10.830472 bpf_flags 0x83,  In
        Juniper PCAP Flags [Ext, no-L2, In], PCAP Extension(s) total length 16
          Device Media Type Extension TLV #3, length 1, value: Ethernet (1)
          Logical Interface Encapsulation Extension TLV #6, length 1, value:
Ethernet (14)
          Device Interface Index Extension TLV #1, length 2, value: 132
          Logical Interface Index Extension TLV #4, length 4, value: 68
        -----original packet-----
        PFE proto 2 (ipv4): (tos 0xc0, ttl 255, id 1511, offset 0, flags [none],
 proto: UDP (17), length: 52) 10.1.1.1.bfd-src > 10.1.1.2.bfd-ip: [udp sum ok]
        BFDv1, length: 24
       One-hop Control, State Up, Flags: [Control Plane Independent], Diagnostic:
 No Diagnostic (0x00)
        Detection Timer Multiplier: 3 (30000 ms Detection time), BFD Length: 24
        My Discriminator: 0x00000002, Your Discriminator: 0x00000001
          Desired min Tx Interval:    10000 ms
          Required min Rx Interval:   10000 ms
          Required min Echo Interval:    0 ms
```

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

Related Documentation

- Understanding Queuing and Marking of Host Outbound Traffic on page 232
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 241
- *monitor traffic*
- *show firewall*
- show interfaces queue on page 1421

Overriding the Input Classification

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

When you override the classification of incoming packets, any mappings you configured for associated precedence bits or incoming interfaces to output transmission queues are

 Copyright © 2019, Juniper Networks, Inc.

ignored. Also, if the packet loss priority (PLP) bit was set in the packet by the incoming interface, the PLP bit is cleared.

To override the input packet classification, do the following:

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

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

   class-name is a name that identifies the routing policy class.

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

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

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

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

Related Documentation

- classification-override on page 1065

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 6

Defining Output Queue Properties with Schedulers

- How Schedulers Define Output Queue Properties on page 254
- Default Schedulers Overview on page 257
- Configuring Schedulers on page 258
- Configuring Scheduler Maps on page 258
- Applying Scheduler Maps Overview on page 259
- Applying Scheduler Maps to Physical Interfaces on page 259
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- Configuring an Input Scheduler on an Interface on page 263
- Understanding Interface Sets on page 264
- Configuring Interface Sets on page 264
- Interface Set Caveats on page 267
- Configuring Internal Scheduler Nodes on page 268
- Example: Configuring and Applying Scheduler Maps on page 269

Copyright © 2019, Juniper Networks, Inc.

How Schedulers Define Output Queue Properties

You use *schedulers* to define the class-of-service (CoS) properties of output queues. You configure CoS properties in a scheduler, then map the scheduler to a forwarding class. Forwarding classes are in turn mapped to output queues. Classifiers map incoming traffic into forwarding classes based on CoS values in well-known packet header fields (behavior aggregate classification) or on multiple packet header fields (multifield classification).

Output queue properties include the amount of interface bandwidth assigned to the queue, the size of the memory buffer allocated for storing packets, the scheduling priority of the queue, and the random early detection (RED) drop profiles associated with the queue to control packet drop during periods of congestion.

Scheduler maps map schedulers to forwarding classes. The output queue mapped to a forwarding class receives the port resources and properties defined in the scheduler mapped to that forwarding class. You apply a scheduler map to an interface to apply queue scheduling to a port. You can associate different scheduler maps with different interfaces to configure port-specific scheduling for forwarding classes (output queues).

To configure class-of-service (CoS) schedulers, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
interfaces {
  interface-name {
    scheduler-map map-name;
    scheduler-map-chassis map-name;
    shaping-rate rate;
    unit {
      output-traffic-control-profile profile-name;
      scheduler-map map-name;
      shaping-rate rate;
    }
  }
}
fabric {
  scheduler-map {
    priority (high | low) scheduler scheduler-name;
  }
}
scheduler-maps {
  map-name {
    forwarding-class class-name scheduler scheduler-name;
  }
}
schedulers {
  scheduler-name {
    buffer-size (percent percentage | remainder | temporal microseconds );
    drop-profile-map loss-priority (any | low | medium-low | medium-high | high)protocol
      (any | non-tcp | tcp) drop-profile profile-name;
    excess-priority (low | high);
    excess-rate percent percentage;
```


Copyright © 2019, Juniper Networks, Inc.

```
    excess-rate (percent percentage | proportion value);
    priority priority-level;
    transmit-rate (rate | percent percentage remainder) <exact | rate-limit>;
  }
}
traffic-control-profiles profile-name {
  delay-buffer-rate (percent percentage | rate);
  excess-rate percent percentage;
  guaranteed-rate (percent percentage | rate);
  scheduler-map map-name;
  shaping-rate (percent percentage | rate);
}
```

You cannot configure both the **shaping-rate** statement at the **[edit class-of-service interfaces *interface-name*]** hierarchy level and the **transmit-rate rate-limit** statement and option at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level. These statements are mutually exclusive. If you do configure both, you will not be able to commit the configuration:

```
[edit class-of-service]
'shaping-rate'
only one option (shaping-rate or transmit-rate rate-limit) can be configured at a time
error: commit failed (statements constraint check failed)
```

NOTE: For PTX Series Packet Transport Routers:

- The **fabric** and **traffic-control-profiles** statements at the **[edit class-of-service]** hierarchy level are not supported.

- Queue Scheduling Components on page 255

Queue Scheduling Components

Table 26 on page 255 provides a quick reference to the scheduler components you can configure to determine the bandwidth properties of output queues (forwarding classes).

Table 26: Output Queue Scheduler Components

Output Queue Scheduler Component	Description
Buffer size	Sets the size of the queue buffer.
Drop profile map	Maps a drop profile to a packet loss priority. Drop profile map components include: • Drop profile—Sets the probability of dropping packets as the queue fills up. • Loss priority—Sets the traffic packet loss priority to which a drop profile applies.

Copyright © 2019, Juniper Networks, Inc.

Table 26: Output Queue Scheduler Components (continued)

Output Queue Scheduler Component	Description
Excess priority	Sets the scheduling priority of excess bandwidth traffic on a scheduler.
Excess rate	Sets the percentage of extra bandwidth (bandwidth that is not used by other queues) a queue can receive. If not set, the device uses the transmit rate to determine how much extra bandwidth the queue can use. Extra bandwidth is the bandwidth remaining after all guaranteed bandwidth requirements are met.
Priority	Sets the scheduling priority applied to the queue.
Shaping rate	Sets a limit on excess bandwidth usage. The transmit rate configures the minimum bandwidth allocated to a queue. Configure the shaping rate as an absolute maximum usage and not the additional usage beyond the configured transmit rate. If you do not set a shaping rate, the default shaping rate is 100 percent, which is the same as no shaping at all.
Transmit rate	Sets the minimum guaranteed bandwidth . By default, if you do not configure an excess rate, extra bandwidth is shared among queues in proportion to the transmit rate of each queue. On strict-high priority queues, sets the amount of bandwidth that receives strict-high priority forwarding treatment. Traffic that exceeds the transmit rate shares in the port excess bandwidth pool based on the strict-high priority excess bandwidth sharing weight of "1", which is not configurable. The actual amount of extra bandwidth that traffic exceeding the transmit rate receives depends on how many other queues consume excess bandwidth and the excess rates of those queues. If you configure two or more strict-high priority queues on a port, you must configure a transmit rate on those queues. However, we strongly recommend that you always configure a transmit rate on strict-high priority queues to prevent them from starving other queues.

Related Documentation

- Understanding How Forwarding Classes Assign Classes to Output Queues on page 205
- Default Schedulers Overview on page 257
- Configuring Schedulers on page 258
- Priority Scheduling Overview on page 327

Copyright © 2019, Juniper Networks, Inc.

Default Schedulers Overview

Each forwarding class has an associated scheduler priority. Only two forwarding classes, best effort and network control (queue 0 and queue 3), are used in the Junos default scheduler configuration.

By default, the best effort forwarding class (queue 0) receives 95 percent of the bandwidth and buffer space for the output link, and the network control forwarding class (queue 3) receives 5 percent. The default drop profile causes the buffer to fill and then discard all packets until it has space.

The expedited-forwarding (queue 1) and assured-forwarding (queue 2) classes have no reserved bandwidth or buffer space because, by default, no schedulers are assigned to those forwarding classes. However, you can manually configure resources for the expedited-forwarding and assured-forwarding classes.

Also by default, each queue can exceed the assigned bandwidth if additional bandwidth is available from other queues. When a forwarding class does not fully use the allocated transmission bandwidth, the remaining bandwidth can be used by other forwarding classes if they receive a larger amount of the offered load than the bandwidth allocated. For more information, see "Allocation of Leftover Bandwidth" on page 284.

The following default scheduler is provided when you install the Junos OS. These settings are not visible in the output of the **show class-of-service** command; rather, they are implicit.

```
[edit class-of-service]
schedulers {
   network-control {
      transmit-rate percent 5;
      buffer-size percent 5;
      priority low;
      drop-profile-map loss-priority any protocol any drop-profile terminal;
   }
   best-effort {
      transmit-rate percent 95;
      buffer-size percent 95;
      priority low;
      drop-profile-map loss-priority any protocol any drop-profile terminal;
   }
}
drop-profiles {
   terminal {
      fill-level 100 drop-probability 100;
   }
}
```

Copyright © 2019, Juniper Networks, Inc.

Configuring Schedulers

You configure a scheduler by including the **scheduler** statement at the **[edit class-of-service]** hierarchy level:

```
schedulers {
  scheduler-name {
    buffer-size (percent percentage | remainder | temporal microseconds);
    drop-profile-map loss-priority (any | low | medium-low | medium-high | high)protocol
      (any | non-tcp | tcp) drop-profile profile-name;
    priority priority-level;
    shaping-rate (percent percentage | rate);
    transmit-rate (rate | percent percentage remainder) <exact | rate-limit>;
  }
}
```

NOTE: Committing changes to schedulers and queues interrupts traffic on affected ports while queue resources are reconfigured.

For detailed information about scheduler configuration statements, see the indicated topics:

- Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size on page 365
- Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 359
- Configuring Scheduler Transmission Rate on page 282
- Configuring Schedulers for Priority Scheduling on page 330

Configuring Scheduler Maps

After defining a scheduler, you can associate it with a specified forwarding class by including it in a *scheduler map*. To do this, include the **scheduler-maps** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
scheduler-maps {
  map-name {
    forwarding-class class-name scheduler scheduler-name;
  }
}
```


Copyright © 2019, Juniper Networks, Inc.

Applying Scheduler Maps Overview

Physical interfaces (for example, **t3-0/0/0**, **t3-0/0/0:0**, and **ge-0/0/0**) support scheduling with any encapsulation type pertinent to that physical interface. For a single port, you cannot apply scheduling to the physical interface if you have applied scheduling to one or more of the associated logical interfaces.

Logical interfaces (for example, **t3-0/0/0 unit 0** and **ge-0/0/0 unit 0**) support scheduling on data link connection identifiers (DLCIs) or VLANs only.

In the Junos OS implementation, the term *logical interfaces* generally refers to interfaces you configure by including the **unit** statement at the **[edit interfaces *interface-name*]** hierarchy level. Logical interfaces have the ***.logical*** descriptor at the end of the interface name, as in **ge-0/0/0.1** or **t1-0/0/0:0.1**, where the logical unit number is **1**.

Although channelized interfaces are generally thought of as logical or virtual, the Junos OS sees T3, T1, and *N*xDS0 interfaces within a channelized IQ PIC as physical interfaces. For example, both **t3-0/0/0** and **t3-0/0/0:1** are treated as physical interfaces by the Junos OS. In contrast, **t3-0/0/0.2** and **t3-0/0/0:1.2** are considered logical interfaces because they have the **.2** at the end of the interface names.

Within the **[edit class-of-service]** hierarchy level, you cannot use the ***.logical*** descriptor when you assign properties to logical interfaces. Instead, you must include the **unit** statement in the configuration. For example:

```
[edit class-of-service]
user@host# set interfaces t3-0/0/0 unit 0 scheduler-map map1
```

Related Documentation

To apply a scheduler map to network traffic, you associate the map with an interface. See the following topics:

- Applying Scheduler Maps to Physical Interfaces on page 259
- Applying Scheduler Maps and Shaping Rate to Physical Interfaces on IQ PICs on page 761
- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300
- Oversubscribing Interface Bandwidth on page 273
- Providing a Guaranteed Minimum Rate on page 285
- Applying Scheduler Maps to Chassis-Level Queues on page 778
- Forwarding Classes and Fabric Priority Queues on page 234
- Associating Schedulers with Fabric Priorities on page 331

Applying Scheduler Maps to Physical Interfaces

After you have defined a scheduler map, as described in "Configuring Scheduler Maps" on page 258, you can apply it to an output interface. Include the **scheduler-map** statement at the **[edit class-of-service interfaces *interface-name*]** hierarchy level:

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces interface-name]
scheduler-map map-name;
```

Interface wildcards are supported. However, scheduler maps using wildcard interfaces are not checked against routing device interfaces at commit time and can result in a configuration that is incompatible with installed hardware. Fully specified interfaces, on the other hand, check the configuration against the hardware and report errors or warning if the hardware does not support the configuration.

Generally, you can associate schedulers with physical interfaces only. For some IQ interfaces, you can also associate schedulers with the logical interface. For more information, see "Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs" on page 300.

NOTE: For original Channelized OC12 PICs, limited CoS functionality is supported. For more information, contact Juniper Networks customer support.

When you apply a scheduler map to a physical interface, or when you modify the configuration of a scheduler map that is already applied to a physical interface, packets already in the output queues of the interface might get dropped. The amount of packet loss is not deterministic and depends on the offered traffic load at the time you apply or modify the scheduler map.

Configuring Traffic Control Profiles for Shared Scheduling and Shaping

Shared scheduling and shaping allows you to allocate separate pools of shared resources to subsets of logical interfaces belonging to the same physical port. You configure shared scheduling and shaping by first creating a traffic-control profile, which specifies a shaping rate and references a scheduler map. You must then share this set of shaping and scheduling resources by applying an instance of the traffic-control profile to a subset of logical interfaces. You can apply a separate instance of the same (or a different) traffic-control profile to another subset of logical interfaces, thereby allocating separate pools of shared resources.

Before you start this procedure:

- Make sure you define a scheduler map. For information about configuring schedulers and scheduler maps, see "Configuring Schedulers" on page 258 and "Configuring Scheduler Maps" on page 258. Gigabit Ethernet IQ2 interfaces support up to eight forwarding classes and queues.

To configure a traffic-control profile, perform the following steps:

1. Create the traffic control profile and configure a shaping rate for it.

```
[edit]
user@host# edit class-of-service traffic-control-profiles profile-name
user@host# set shaping-rate (percent percentage | rate)
```


Copyright © 2019, Juniper Networks, Inc.

You can configure the shaping rate as a percentage from 1 through 100 or as an absolute rate from 1000 through 6,400,000,000,000 bits per second (bps). The shaping rate corresponds to a peak information rate (PIR). For more information, see "Oversubscribing Interface Bandwidth" on page 273.

2. Define an association between the traffic-control profile and a previously configured scheduler map by including the **scheduler-map** statement at the **[edit class-of-service traffic-control-profiles *profile-name*]** hierarchy level.

```
[edit class-of-service traffic-control-profiles profile-name]
user@host# set scheduler-map map-name;
```

3. 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 class-of-service traffic-control-profiles profile-name]
user@host# set delay-buffer-rate (percent percentage | rate)
```

   You can configure the **delay-buffer** rate as a percentage from 1 through 100 or as an absolute rate from 1000 through 6,400,000,000,000 bits per second. The delay-buffer rate controls latency. For more information, see "Oversubscribing Interface Bandwidth" on page 273 and "Providing a Guaranteed Minimum Rate" on page 285.

4. Configure a guaranteed minimum rate for the traffic-control profile.

```
[edit class-of-service traffic-control-profiles profile-name]
user@host# set guaranteed-rate (percent percentage | rate)
```

   You can configure the guaranteed rate as a percentage from 1 through 100 or as an absolute rate from 1000 through 6,400,000,000,000 bps. The guaranteed rate corresponds to a committed information rate (CIR). For more information, see "Providing a Guaranteed Minimum Rate" on page 285.

   You must now share an instance of the traffic-control profile.

5. Enable shared-scheduling on the interface.

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

   This statement enables logical interfaces belonging to the same physical port to share one set of shaping and scheduling resources.

Copyright © 2019, Juniper Networks, Inc.

NOTE: On each physical interface, the **shared-scheduler** and **per-unit-scheduler** statements are mutually exclusive. Even so, you can configure one logical interface for each shared instance. This effectively provides the functionality of per-unit scheduling.

6. (Optional) Apply the traffic-control profile to an input interface.

```
[edit]
user@host# edit class-of-service interfaces interface-name unit logical-unit-number
user@host# set input-traffic-control-profile profile-name shared-instance
  instance-name
```

These statements are explained in Step 7.

7. (Optional) Apply the traffic-control profile to an output interface.

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

The profile name references the traffic-control profile you configured in Step 1 through Step 4. The **shared-instance** name does not reference a configuration. It can be any text string you wish to apply to multiple logical interfaces that you want to share the set of resources configured in the traffic-control profile. Each logical interface shares a set of scheduling and shaping resources with other logical interfaces that are on the same physical port and that have the same shared-instance name applied.

This concept is demonstrated in "Example: Configuring Shared Resources on Ethernet IQ2 Interfaces" on page 832.

NOTE: You cannot include the **output-traffic-control-profile** statement in the configuration if either the **scheduler-map** or **shaping-rate** statement is included in the logical interface configuration.

Related Documentation

- Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels on page 576
- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300
- Hierarchical Schedulers and Traffic Control Profiles on page 343

Copyright © 2019, Juniper Networks, Inc.

Configuring an Input Scheduler on an Interface

As an alternative to shared input traffic-control profiles, you can configure each interface to use its own input scheduler. For each physical interface, you can apply an input scheduler map to the physical interface or its logical interfaces, but not both.

Input scheduler maps are supported on the following Ethernet interfaces:

- IQ2 and IQ2E PICs
- DPCs and MPCs that support Enhanced Queuing (Q/EQ)
- MX80 with support for per-VLAN queuing

Before you start this procedure:

- Define a scheduler map at the **[edit class-of-service scheduler-maps]** hierarchy level.

To configure a separate input scheduler on the physical interface:

- Specify the name of the physical interface and the scheduler map.

```
[edit class-of-service interfaces interface-name]
user@host# set input-scheduler-map map-name
```

To configure a separate input scheduler on a logical interface:

1. Specify the name of the physical and logical interface and scheduler map.

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

2. Enable the association of the scheduler map name and interface.

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

The **per-unit-scheduler** statement enables one set of output queues for each logical interface configured under the physical interface.

On Gigabit Ethernet IQ2 PIC interfaces, configuration of the **per-unit-scheduler** statement requires that you configure VLAN tagging also. When you include the **per-unit-scheduler** statement, the maximum number of VLANs supported is 768 on a single-port Gigabit Ethernet IQ PIC. On a dual-port Gigabit Ethernet IQ PIC, the maximum number is 384.

Related Documentation

- Configuring Schedulers on page 258
- Configuring Scheduler Maps on page 258

Copyright © 2019, Juniper Networks, Inc.

- input-traffic-control-profile on page 1177

Understanding Interface Sets

Although the interface set is applied at the **[edit interfaces]** hierarchy level, the CoS parameters for the interface set are defined at the **[edit class-of-service interfaces]** hierarchy level, usually with the **output-traffic-control-profile** *profile-name* statement.

This example applies a traffic control profile called **tcp-set1** to an interface set called **set-ge-0**:

```
[edit class-of-service interfaces]
interface-set set-ge-0 {
   output-traffic-control-profile tcp-set1;
}
```

Related Documentation

- output-traffic-control-profile on page 1219

Configuring Interface Sets

To configure an interface set, include the **interface-set** statement at the **[edit class-of-service interfaces]** hierarchy level:

```
[edit class-of-service interfaces]
interface-set interface-set-name {
   ...interface-cos-configuration-statements ...
}
```

To apply the interface set to interfaces, include the **interface-set** statement at the **[edit interfaces]** hierarchy level:

```
[edit interfaces]
interface-set interface-set-name {
   interface ethernet-interface-name {
      ... interface-cos-configuration-statements ...
   }
}
```

Interface sets can be defined in two major ways:

- As a list of logical interfaces or aggregated Ethernet interfaces (**unit 100**, **unit 200**, and so on)
- At the stacked VLAN level using a list of outer VLAN IDs (**vlan-tags-outer 210**, **vlan-tags-outer 220**, and so on).

  The **svlan** *number* listing option with a single outer VLAN tag is a convenient way to specify a set of VLAN members having the same outer VLAN tags. Service providers can use these statements to group interfaces to apply scheduling parameters such as guaranteed rate and shaping rate to the traffic in the groups.

Copyright © 2019, Juniper Networks, Inc.

Whether using the logical interface listing option for a group of customer VLANs, aggregated Ethernet interfaces, or the S-VLAN set listing option for a group of VLAN outer tags, all traffic heading downstream must be gathered into an interface set with the **interface-set** statement at the **[edit class-of-service interfaces]** hierarchy level.

Regardless of listing convention, you can only use one of the types in an interface set. Examples of this limitation appear later in this section.

NOTE: Interface sets are currently used only by CoS, but they are applied at the **[edit interfaces]** hierarchy level to make them available to other services that might use them in future.

```
[edit interfaces]
interface-set interface-set-name {
   interface ethernet-interface-name {
      (unit logical-unit-number | vlan-tags-outer vlan-tag) {
         ...
      }
   }
}
```

The logical interface naming option lists Ethernet interfaces:

```
[edit interfaces]
interface-set unitl-set-ge-0 {
   interface ge-0/0/0 {
      unit 0;
      unit 1;
      ...
   }
}
```

The interface naming option lists aggregated Ethernet interfaces:

```
[edit interfaces]
   interface-set demuxset1 {
      interface demux0 {
         unit 1;
         ..
      }
   }
   demux0 {
      unit 1 {
         demux-options {
            underlying-interface ae0.1;
         }
         family inet {
            demux-source {
               10.1.1.1/24;
            }
            address 10.1.1.1/24;
         }
```

Copyright © 2019, Juniper Networks, Inc.

```
    }
    ..
    ae0 {
      unit 1 {
      }
      ..
    }
  }
  class-of-service {
    interface-set demuxset1 {
      output-traffic-control-profile tcp2;
    }
  }
}
```

The S-VLAN option lists only one S-VLAN (outer) tag value:

```
[edit interfaces]
interface-set svlan-set {
  interface ge-1/0/0 {
    vlan-tags-outer 2000;
  }
}
```

The S-VLAN naming option lists S-VLAN (outer) tag values:

```
[edit interfaces]
interface-set svlan-set-tags {
  interface ge-2/0/0 {
    vlan-tags-outer 2000;
    vlan-tags-outer 2001;
    vlan-tags-outer 2002;
    ...
  }
}
```

NOTE: Ranges are not supported: you must list each VLAN or logical interface separately.

Related Documentation

- Interface Set Caveats on page 267

Copyright © 2019, Juniper Networks, Inc.

Interface Set Caveats

When configuring interface sets, consider the following guidelines:

- Interface sets can be defined in two major ways: as a list of logical interfaces or groups of aggregated Ethernet logical interfaces (**unit 100**, **unit 200**, and so on), or at the stacked VLAN level using a list of outer VLAN IDs (**vlan-tags-outer 210**, **vlan-tags-outer 220**, and so on). You can configure sets of aggregated Ethernet interfaces on MIC or MPC interfaces only.
- You cannot specify an interface set mixing the logical interface, aggregated Ethernet, S-VLAN, or VLAN outer tag list forms of the **interface-set** statement.
- Keep the following guidelines in mind when configuring interface sets of logical interfaces over aggregated Ethernet:
  - Sets of aggregated Ethernet interfaces are supported on MIC and MPC interfaces 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.
- A logical interface can only belong to one interface set. If you try to add the same logical interface to different interface sets, the commit operation fails.

  This example generates a commit error:

```
[edit interfaces]
interface-set set-one {
   interface ge-2/0/0 {
      unit 0;
      unit 2;
   }
}
interface-set set-two {
   interface ge-2/0/0 {
      unit 1;
      unit 3;
      unit 0; # COMMIT ERROR! Unit 0 already belongs to set-one.
   }
}
```

Copyright © 2019, Juniper Networks, Inc.

- Members of an interface set cannot span multiple physical interfaces. Only one physical interface is allowed to appear in an interface set.

  This configuration is not supported:

```
[edit interfaces]
interface-set set-group {
   interface ge-0/0/1 {
      unit 0;
      unit 1;
   }
   interface ge-0/0/2 { # This is NOT supported in the same interface set!
      unit 0;
      unit 1;
   }
}
```

Related Documentation

- Configuring Interface Sets on page 264

Configuring Internal Scheduler Nodes

A node in the hierarchy is considered internal if either of the following conditions apply:

- Any one of its children nodes has a traffic control profile configured and applied.
- You include the **internal-node** statement at the **[edit class-of-service interfaces interface-set *set-name*]** hierarchy level.

Why would it be important to make a certain node internal? Generally, there are more resources available at the logical interface (unit) level than at the interface set level. Also, it might be desirable to configure all resources at a single level, rather than spread over several levels. The **internal-node** statement provides this flexibility. This can be a helpful configuration device when interface-set queuing without logical interfaces is used exclusively on the interface.

The **internal-node** statement can be used to raise the interface set without children to the same level as the other configured interface sets with children, allowing them to compete for the same set of resources.

In summary, using the **internal-node** statement allows statements to all be scheduled at the same level with or without children.

The following example makes the interfaces sets **if-set-1** and **if-set-2** internal:

```
[edit class-of-service interfaces]
interface-set {
   if-set-1 {
      internal-node;
      output-traffic-control-profile tcp-200m-no-smap;
   }
   if-set-2 {
```


Copyright © 2019, Juniper Networks, Inc.

```
    internal-node;
    output-traffic-control-profile tcp-100m-no-smap;
  }
}
```

If an interface set has logical interfaces configured with a traffic control profile, then the use of the **internal-node** statement has no effect.

Internal nodes can specify a **traffic-control-profile-remaining** statement.

Example: Configuring and Applying Scheduler Maps

This example shows how to configure and apply a scheduler map to a device's interface.

- Requirements on page 269
- Overview on page 269
- Configuration on page 270
- Verification on page 271

Requirements

Before you begin:

- Create and configure the forwarding classes. See "Configuring a Custom Forwarding Class for Each Queue" on page 212.
- Create and configure the schedulers. See *Example: Configuring Class-of-Service Schedulers on a Security Device*.

Overview

After you define a scheduler, you can include it in a scheduler map, which maps a specified forwarding class to a scheduler configuration. You configure a scheduler map to assign a forwarding class to a scheduler, and then apply the scheduler map to any interface that must enforce DiffServ CoS.

After they are applied to an interface, the scheduler maps affect the hardware queues, packet schedulers, and RED drop profiles.

In this example, you create the scheduler map diffserv-cos-map and apply it to the device's Ethernet interface ge-0/0/0. The map associates the mf-classifier forwarding classes to the schedulers as shown in Table 27 on page 269.

Table 27: Sample diffserv-cos-map Scheduler Mapping

mf-classifier Forwarding Class	For CoS Traffic Type	diffserv-cos-map Scheduler
be-class	Best-effort traffic	be-scheduler
ef-class	Expedited forwarding traffic	ef-scheduler
af-class	Assured forwarding traffic	af-scheduler

Copyright © 2019, Juniper Networks, Inc.

Table 27: Sample diffserv-cos-map Scheduler Mapping (continued)

mf-classifier Forwarding Class	For CoS Traffic Type	diffserv-cos-map Scheduler
nc-class	Network control traffic	nc-scheduler

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, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter **commit** from the configuration mode.

```
set class-of-service scheduler-maps diffserv-cos-map forwarding-class be-class scheduler
    be-scheduler
set class-of-service scheduler-maps diffserv-cos-map forwarding-class ef-class scheduler
    ef-scheduler
set class-of-service scheduler-maps diffserv-cos-map forwarding-class af-class scheduler
    af-scheduler
set class-of-service scheduler-maps diffserv-cos-map forwarding-class nc-class scheduler
    nc-scheduler
set class-of-service interfaces ge-0/0/0 unit 0 scheduler-map diffserv-cos-map
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure and apply a scheduler map to a device's interface:

1. Configure a scheduler map for DiffServ CoS.

   ```
   [edit class-of-service]
   user@host# edit scheduler-maps diffserv-cos-map
   ```

2. Configure a best-effort forwarding class and scheduler.

   ```
   [edit class-of-service scheduler-maps diffserv-cos-map]
   user@host# set forwarding-class be-class scheduler be-scheduler
   ```

3. Configure an expedited forwarding class and scheduler.

   ```
   [edit class-of-service scheduler-maps diffserv-cos-map]
   user@host# set forwarding-class ef-class scheduler ef-scheduler
   ```

4. Configure an assured forwarding class and scheduler.

   ```
   [edit class-of-service scheduler-maps diffserv-cos-map]
   user@host# set forwarding-class af-class scheduler af-scheduler
   ```


Copyright © 2019, Juniper Networks, Inc.

5. Configure a network control class and scheduler.

```
[edit class-of-service scheduler-maps diffserv-cos-map]
user@host# set forwarding-class nc-class scheduler nc-scheduler
```

6. Apply the scheduler map to an interface.

```
[edit class-of-service]
user@host# set interfaces ge-0/0/0 unit 0 scheduler-map diffserv-cos-map
```

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

```
[edit]
user@host# show class-of-service
interfaces {
   ge-0/0/0 {
      unit 0 {
      scheduler-map diffserv-cos-map;
   }
   }
}
scheduler-maps {
   diffserv-cos-map {
      forwarding-class be-class scheduler be-scheduler;
      forwarding-class ef-class scheduler ef-scheduler;
      forwarding-class af-class scheduler af-scheduler;
      forwarding-class nc-class scheduler nc-scheduler;
   }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Verifying the Scheduler Map Configuration

Purpose Verify that scheduler maps are configured properly.

Action From operational mode, enter the **show class-of-service** command.

Related Documentation

- Default Schedulers Overview on page 257
- Configuring Schedulers on page 258
- Configuring Scheduler Maps on page 258

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 7

Controlling Bandwidth with Scheduler Rates

- Oversubscribing Interface Bandwidth on page 273
- Configuring Scheduler Transmission Rate on page 282
- Providing a Guaranteed Minimum Rate on page 285
- PIR-Only and CIR Mode on page 290
- Excess Rate and Excess Priority Configuration Examples on page 290
- Controlling Remaining Traffic on page 295
- Bandwidth Sharing on Nonqueuing Packet Forwarding Engines Overview on page 298
- Configuring Rate Limits on Nonqueuing Packet Forwarding Engines on page 299
- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300
- Example: Applying Scheduler Maps and Shaping Rate to DLCIs on page 308
- Example: Applying Scheduling and Shaping to VLANs on page 312
- Example: Limiting Egress Traffic on an Interface Using Port Shaping for CoS on page 319
- Configuring Input Shaping Rates for Both Physical and Logical Interfaces on page 326

Oversubscribing Interface Bandwidth

The term *oversubscribing interface bandwidth* means configuring shaping rates (peak information rates [PIRs]) so that their sum exceeds the interface bandwidth.

On Channelized IQ PICs, Gigabit Ethernet IQ PICs, and FRF.15 and FRF.16 link services IQ (LSQ) interfaces on Services PICs, Multiservices PICs, and Multiservices DPCs, you can oversubscribe interface bandwidth. This means that the logical interfaces (and DLCIs within an FRF.15 or FRF.16 bundle) can be oversubscribed when there is leftover bandwidth. In the case of FRF.16 bundle interfaces, the physical interface can be oversubscribed. The oversubscription is capped to the configured PIR. Any unused bandwidth is distributed equally among oversubscribed logical interfaces or data-link connection identifiers (DLCIs), or physical interfaces.

For networks that are not likely to experience congestion, oversubscribing interface bandwidth improves network utilization, thereby allowing more customers to be provisioned on a single interface. If the actual data traffic does not exceed the interface

Copyright © 2019, Juniper Networks, Inc.

bandwidth, oversubscription allows you to sell more bandwidth than the interface can support.

We recommend avoiding oversubscription in networks that are likely to experience congestion. Be cautious not to oversubscribe a service by too much, because this can cause degradation in the performance of the routing platform during congestion. When you configure oversubscription, starvation of some output queues can occur if the actual data traffic exceeds the physical interface bandwidth. You can prevent degradation by using statistical multiplexing to ensure that the actual data traffic does not exceed the interface bandwidth.

NOTE: You cannot oversubscribe interface bandwidth when you configure traffic shaping using the method described in "Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs" on page 300.

When configuring oversubscription for FRF.16 bundle interfaces, you can assign traffic control profiles that apply on a physical interface basis. When you apply traffic control profiles to FRF.16 bundles at the *logical* interface level, member link interface bandwidth is underutilized when there is a small proportion of traffic or no traffic at all on an individual DLCI. Support for traffic control features on the FRF.16 bundle physical interface level addresses this limitation.

To configure oversubscription of the interface, perform the following steps:

1. Include the **shaping-rate** statement at the **[edit class-of-service traffic-control-profiles *profile-name*]** hierarchy level:

   ```
   [edit class-of-service traffic-control-profiles profile-name]
   shaping-rate (percent percentage | rate);
   ```

   NOTE: When configuring oversubscription for FRF.16 bundle interfaces on a physical interface basis, you *must* specify shaping-rate as a percentage.

   On LSQ interfaces, you can configure the shaping rate as a percentage from 1 through 100.

   On IQ and IQ2 interfaces, you can configure the shaping rate as an absolute rate from 1000 through 6,400,000,000,000 bps.

   For all MX Series router and EX Series switch interfaces, the shaping rate can be from 65,535 through 6,400,000,000,000 bps.

   Alternatively, you can configure a shaping rate for a logical interface and oversubscribe the physical interface by including the **shaping-rate** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level. However, with this configuration approach, you cannot independently control the delay-buffer rate, as described in Step 2.

Copyright © 2019, Juniper Networks, Inc.

NOTE: For channelized and Gigabit Ethernet IQ interfaces, the **shaping-rate** and **guaranteed-rate** statements are mutually exclusive. You cannot configure some logical interfaces to use a shaping rate and others to use a guaranteed rate. This means there are no service guarantees when you configure a PIR. For these interfaces, you can configure either a PIR or a committed information rate (CIR), but not both.

This restriction does not apply to Gigabit Ethernet IQ2 PICs or LSQ interfaces on Multiservices and Services PICs. For LSQ and Gigabit Ethernet IQ2 interfaces, you can configure both a PIR and a CIR on an interface. For more information about CIRs, see "Providing a Guaranteed Minimum Rate" on page 285.

For more information about Gigabit Ethernet IQ2 PICs, see "CoS on Enhanced IQ2 PICs Overview" on page 794.

2. Optionally, you can base the delay-buffer calculation on a delay-buffer rate. To do this, include the **delay-buffer-rate** statement at the **[edit class-of-service traffic-control-profiles *profile-name*]** hierarchy level:

NOTE: When configuring oversubscription for FRF.16 bundle interfaces on a physical interface basis, you *must* specify **delay-buffer-rate** as a percentage.

```
[edit class-of-service traffic-control-profiles profile-name]
delay-buffer-rate (percent percentage | rate);
```

The delay-buffer rate overrides the shaping rate as the basis for the delay-buffer calculation. In other words, the shaping rate or scaled shaping rate is used for delay-buffer calculations only when the delay-buffer rate is not configured.

For LSQ interfaces, if you do not configure a delay-buffer rate, the guaranteed rate (CIR) is used to assign buffers. If you do not configure a guaranteed rate, the shaping rate (PIR) is used in the undersubscribed case, and the scaled shaping rate is used in the oversubscribed case.

On LSQ interfaces, you can configure the delay-buffer rate as a percentage from 1 through 100.

On IQ and IQ2 interfaces, you can configure the delay-buffer rate as an absolute rate from 1000 through 6,400,000,000,000 bps.

The actual delay buffer is based on the calculations described in "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365. For an example showing how the delay-buffer rates are applied, see "Examples: Oversubscribing Interface Bandwidth" on page 279.

Copyright © 2019, Juniper Networks, Inc.

Configuring large buffers on relatively slow-speed links can cause packet aging. To help prevent this problem, the software requires that the sum of the delay-buffer rates be less than or equal to the port speed.

This restriction does not eliminate the possibility of packet aging, so you should be cautious when using the **delay-buffer-rate** statement. Though some amount of extra buffering might be desirable for burst absorption, delay-buffer rates should not far exceed the service rate of the logical interface.

If you configure delay-buffer rates so that the sum exceeds the port speed, the configured delay-buffer rate is not implemented for the last logical interface that you configure. Instead, that logical interface receives a delay-buffer rate of zero, and a warning message is displayed in the CLI. If bandwidth becomes available (because another logical interface is deleted or deactivated, or the port speed is increased), the configured delay-buffer-rate is reevaluated and implemented if possible.

If you do not configure a delay-buffer rate or a guaranteed rate, the logical interface receives a delay-buffer rate in proportion to the shaping rate and the remaining delay-buffer rate available. In other words, the delay-buffer rate for each logical interface with no configured delay-buffer rate is equal to:

```
(remaining delay-buffer rate * shaping rate) / (sum of shaping rates)
```

where the remaining delay-buffer rate is equal to:

```
(interface speed) - (sum of configured delay-buffer rates)
```

3. To assign a scheduler map to the logical interface, include the **scheduler-map** statement at the **[edit class-of-service traffic-control-profiles *profile-name*]** hierarchy level:

```
[edit class-of-service traffic-control-profiles profile-name]
scheduler-map map-name;
```

For information about configuring schedulers and scheduler maps, see "Configuring Schedulers" on page 258 and "Configuring Scheduler Maps" on page 258.

4. Optionally, you can enable large buffer sizes to be configured. To do this, include the **q-pic-large-buffer** statement at the **[edit chassis fpc *slot-number* pic *pic-number*]** hierarchy level:

```
[edit chassis fpc slot-number pic pic-number]
q-pic-large-buffer;
```

If you do not include this statement, the delay-buffer size is more restricted. We recommend restricted buffers for delay-sensitive traffic, such as voice traffic. For more information, see "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365.

5. To enable scheduling on logical interfaces, include the **per-unit-scheduler** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
```


Copyright © 2019, Juniper Networks, Inc.

```
per-unit-scheduler;
```

When you include this statement, the maximum number of VLANs supported is 768 on a single-port Gigabit Ethernet IQ PIC. On a dual-port Gigabit Ethernet IQ PIC, the maximum number is 384.

6. To enable scheduling for FRF.16 bundles physical interfaces, include the **no-per-unit-scheduler** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
no-per-unit-scheduler;
```

7. To apply the traffic-scheduling profile , include the output-traffic-control-profile statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
output-traffic-control-profile profile-name;
```

You cannot include the **output-traffic-control-profile** statement in the configuration if either the **scheduler-map** or **shaping-rate** statement is included in the logical interface configuration.

Table 28 on page 277 shows how the bandwidth and delay buffer are allocated in various configurations.

Table 28: Bandwidth and Delay Buffer Allocations by Configuration Scenario

Configuration Scenario	Delay Buffer Allocation
You do not oversubscribe the interface. You do not configure a guaranteed rate. You do not configure a shaping rate. You do not configure a delay-buffer rate.	Logical interface receives the remaining bandwidth and receives a delay buffer in proportion to the remaining bandwidth.
You do not oversubscribe the interface. You configure a shaping rate at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.	For backward compatibility, the shaped logical interface receives a delay buffer based on the shaping rate. The multiplicative factor depends on whether you include the **q-pic-large-buffer** statement. For more information, see "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365. Unshaped logical interfaces receive the remaining bandwidth and a delay buffer in proportion to the remaining bandwidth.
You oversubscribe the interface. You do not configure a guaranteed rate. You do not configure a shaping rate. You do not configure a delay-buffer rate.	Logical interface receives minimal bandwidth with no guarantees and receives a minimal delay buffer equal to four MTU-sized packets.

Copyright © 2019, Juniper Networks, Inc.

Table 28: Bandwidth and Delay Buffer Allocations by Configuration Scenario (continued)

Configuration Scenario	Delay Buffer Allocation
You oversubscribe the interface. You configure a shaping rate. You do not configure a guaranteed rate. You do not configure a delay-buffer rate.	Logical interface receives a delay buffer based on the scaled shaping rate: **scaled shaping rate = (shaping-rate * [physical interface bandwidth]) / SUM (shaping-rates of all logical interfaces on the physical interface)** The logical interface receives variable bandwidth, depending on how much oversubscription and statistical multiplexing is present. If the amount of oversubscription is low enough that statistical multiplexing does not make all logical interfaces active at the same time and the physical interface bandwidth is not exceeded, the logical interface receives bandwidth equal to the shaping rate. Otherwise, the logical interface receives a smaller amount of bandwidth. In either case, the logical interface bandwidth does not exceed the shaping rate.
You oversubscribe the interface. You configure a shaping rate. You configure a delay-buffer rate.	Logical interface receives a delay buffer based on the delay-buffer rate. For example, on IQ and IQ2 interfaces: **delay-buffer-rate <= 10 Mbps: 400-millisecond (ms) delay buffer** **delay-buffer-rate <= 20 Mbps: 300-ms delay buffer** **delay-buffer-rate <= 30 Mbps: 200-ms delay buffer** **delay-buffer-rate <= 40 Mbps: 150-ms delay buffer** **delay-buffer-rate > 40 Mbps: 100-ms delay buffer** On LSQ DLCIs, if **total bundle bandwidth < T1 bandwidth**: **delay-buffer-rate = 1 second** On LSQ DLCIs, if **total bundle bandwidth >= T1 bandwidth**: **delay-buffer-rate = 200 ms** The multiplicative factor depends on whether you include the **q-pic-large-buffer** statement. For more information, see "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365. The logical interface receives variable bandwidth, depending on how much oversubscription and statistical multiplexing is present. If the amount of oversubscription is low enough that statistical multiplexing does not make all logical interfaces active at the same time and the physical interface bandwidth is not exceeded, the logical interface receives bandwidth equal to the shaping rate. Otherwise, the logical interface receives a smaller amount of bandwidth. In either case, the logical interface bandwidth does not exceed the shaping rate.
You oversubscribe the interface. You do not configure a shaping rate. You configure a guaranteed rate. You configure a delay-buffer rate.	Logical interface receives a delay buffer based on the delay-buffer rate.
You oversubscribe the interface. You do not configure a shaping rate. You do not configure a guaranteed rate. You configure a delay-buffer rate.	This scenario is not allowed. If you configure a delay-buffer rate, the traffic-control profile must also include either a shaping rate or a guaranteed rate.
You oversubscribe the interface. You configure a shaping rate. You configure a guaranteed rate. You do not configure a delay-buffer rate.	Logical interface receives a delay buffer based on the guaranteed rate. This configuration is valid on LSQ interfaces and Gigabit Ethernet IQ2 interfaces only. On channelized interfaces, you cannot configure both a shaping rate (PIR) and a guaranteed rate (CIR).

Copyright © 2019, Juniper Networks, Inc.

NOTE: In Junos OS Release 13.3, IP packets with DLCI 0 or 1023 are identified as part of control traffic and routed to the high-priority queue. This oversubscribes the high-priority queue, which is reserved for frame relay control traffic. Oversubscribing the high-priority queue causes the frame relay Local Management Interface (LMI) packets to be dropped.

Verifying Configuration of Bandwidth Oversubscription

To verify your configuration, you can issue this following operational mode commands:

- **show class-of-service interfaces**
- **show class-of-service traffic-control-profile** ***profile-name***

Examples: Oversubscribing Interface Bandwidth

This section provides two examples: oversubscription of a channelized interface and oversubscription of an LSQ interface.

Oversubscribing a Channelized Interface

Two logical interface units, **0** and **1**, are shaped to rates 2 Mbps and 3 Mbps, respectively. The delay-buffer rates are 750 Kbps and 500 Kbps, respectively. The actual delay buffers allocated to each logical interface are 1 second of 750 Kbps and 2 seconds of 500 Kbps, respectively. The 1-second and 2-second values are based on the following calculations:

```
delay-buffer-rate < [16 x 64 Kbps]): 1 second of delay-buffer-rate
delay-buffer-rate < [8 x 64 Kbps]): 2 seconds of delay-buffer-rate
```

For more information about these calculations, see "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365.

```
chassis {
   fpc 3 {
      pic 0 {
         q-pic-large-buffer;
      }
   }
}
interfaces {
   t1-3/0/0 {
      per-unit-scheduler;
   }
}
class-of-service {
   traffic-control-profiles {
      tc-profile1 {
         shaping-rate 2m;
         delay-buffer-rate 750k; # 750 Kbps is less than 16 x 64 Kbps
         scheduler-map sched-map1;
      }
      tc-profile2 {
         shaping-rate 3m;
```

Copyright © 2019, Juniper Networks, Inc.

```
        delay-buffer-rate 500k; # 500 Kbps is less than 8 x 64 Kbps
        scheduler-map sched-map2;
      }
    }
    interfaces {
      t1-3/0/0 {
        unit 0 {
          output-traffic-control-profile tc-profile1;
        }
        unit 1 {
          output-traffic-control-profile tc-profile2;
        }
      }
    }
}
```

Oversubscribing an LSQ Interface with Scheduling Based on the Logical Interface

Apply a traffic-control profile to a logical interface representing a DLCI on an FRF.16 bundle:

```
interfaces {
  lsq-1/3/0:0 {
    per-unit-scheduler;
    unit 0 {
      dlci 100;
    }
    unit 1 {
      dlci 200;
    }
  }
}

class-of-service {
  traffic-control-profiles {
    tc_0 {
      shaping-rate percent 100;
      guaranteed-rate percent 60;
      delay-buffer-rate percent 80;
    }
    tc_1 {
      shaping-rate percent 80;
      guaranteed-rate percent 40;
    }
  }
  interfaces {
    lsq-1/3/0 {
      unit 0 {
        output-traffic-control-profile tc_0;
      }
      unit 1 {
        output-traffic-control-profile tc_1;
      }
    }
  }
```


Copyright © 2019, Juniper Networks, Inc.

```
}
```

Oversubscribing an LSQ Interface with Scheduling Based on the Physical Interface

Apply a traffic-control profile to the physical interface representing an FRF.16 bundle:

```
interfaces {
   lsq-0/2/0:0 {
      no-per-unit-scheduler;
      encapsulation multilink-frame-relay-uni-nni;
      unit 0 {
         dlci 100;
         family inet {
            address 10.18.18.2/24;
         }
      }
   }
class-of-service {
   traffic-control-profiles {
      rlsq_tc {
         scheduler-map rlsq;
         shaping-rate percent 60;
         delay-buffer-rate percent 10;
      }
   }
   interfaces {
      lsq-0/2/0:0 {
         output-traffic-control-profile rlsq_tc;
      }
   }
}
scheduler-maps {
   rlsq {
      forwarding-class best-effort scheduler rlsq_scheduler;
      forwarding-class expedited-forwarding scheduler rlsq_scheduler1;
   }
}
schedulers {
   rlsq_scheduler {
      transmit-rate percent 20;
      priority low;
   }
   rlsq_scheduler1 {
      transmit-rate percent 40;
      priority high;
   }
}
```

On an FRF.15 bundle, apply the following configuration:

```
class-of-service {
   traffic-control-profiles {
      rlsq {
         scheduler-map sched_0;
         shaping-rate percent 40;
```

Copyright © 2019, Juniper Networks, Inc.

```
        delay-buffer-rate percent 50;
      }
    }
    interfaces lsq-2/0/0 {
      unit 0 {
        output-traffic-control-profile rlsq;
      }
    }
}
interfaces lsq-2/0/0 {
    per-unit-scheduler;
    unit 0 {
      encapsulation multilink-frame-relay-end-to-end;
      family inet {
        address 10.1.1.2/32;
      }
    }
}
```

Configuring Scheduler Transmission Rate

The transmission rate control determines the actual traffic bandwidth from each forwarding class you configure. The rate is specified in bits per second (bps). Each queue is allocated some portion of the bandwidth of the outgoing interface.

This bandwidth amount can be a fixed value, such as 1 megabit per second (Mbps), a percentage of the total available bandwidth, or the rest of the available bandwidth. You can limit the transmission bandwidth to the exact value you configure, or allow it to exceed the configured rate if additional bandwidth is available from other queues. This property allows you to ensure that each queue receives the amount of bandwidth appropriate to its level of service.

On M Series routers other than the M120 and M320 routers, you should not configure a **buffer-size** larger than the **transmit-rate** for a rate-limited queue in a scheduler. If you do, the Packet Forwarding Engine will reject the CoS configuration. However, you can achieve the same effect by removing the **exact** option from the transmit rate or specifying the buffer size using the **temporal** option.

NOTE: For 8-port, 12-port, and 48-port Fast Ethernet PICs, transmission scheduling is not supported.

To configure transmission scheduling, include the **transmit-rate** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level:

```
[edit class-of-service schedulers scheduler-name]
transmit-rate (rate | percent percentage | remainder) <exact | rate-limit>;
```

You can specify the transmit rate as follows:

Copyright © 2019, Juniper Networks, Inc.

- ***rate***—Transmission rate, in bits per second. For all MX Series router and EX Series switch interfaces, the rate can be from 65,535 through 6,400,000,000,000 bps. On all other platforms, the rate can be from 3200 through 6,400,000,000,000 bps.
- **percent *percentage***—Percentage of transmission capacity.
- **remainder**—Use remaining rate available. In the configuration, you cannot combine the **remainder** and **exact** options.
- **exact**—(Optional) Enforce the exact transmission rate or percentage you configure with the **transmit-rate *rate*** or **transmit-rate percent** statement. Under sustained congestion, a rate-controlled queue that goes into negative credit fills up and eventually drops packets. You specify the **exact** option as follows:

```
[edit class-of-service schedulers scheduler-name]
transmit-rate rate exact;

[edit class-of-service schedulers scheduler-name]
transmit-rate percent percentage exact;
```

In the configuration, you cannot combine the **remainder** and **exact** options.

NOTE:

- Including the exact option is not supported on Enhanced Queuing Dense Port Concentrators (DPCs) on Juniper Network MX Series 5G Universal Routing Platforms.
- The configuration of the transmit-rate percent 0 exact statement at the [edit class-of-service schedulers *scheduler-name*] hierarchy is ineffective on T4000 routers with Type 5 FPCs.

Copyright © 2019, Juniper Networks, Inc.

- **rate-limit**—(Optional) Limit the transmission rate to the specified amount. You can configure this option for all 8 queues of a logical interface (unit) and apply it to shaped or unshaped logical interfaces. If you configure a zero rate-limited transmit rate, all packets belonging to that queue are dropped. On IQE PICs, the **rate-limit** option for the schedulers' transmit rate is implemented as a static policer. Therefore, these schedulers are not aware of congestion and the maximum rate possible on these schedulers is limited by the value specified in the **transmit-rate** statement. Even if there is no congestion, the queue cannot send traffic above the transmit rate due to the static policer.

NOTE: You can apply a transmit rate limit to logical interfaces on Multiservices 100, 400, or 500 PICs. Typically, rate limits are used to prevent a strict-high queue (such as voice) from starving lower priority queues. You can only rate-limit one queue per logical interface. To apply a rate-limit to a Multiservices PIC interface, configure the rate limit in a scheduler and apply the scheduler map to the Multiservices (lsq-) interface at the **[edit class-of-service interfaces]** hierarchy level. For information about configuring other scheduler components, see "Configuring Schedulers" on page 258.

For more information about scheduler transmission rate, see the following sections:

- Example: Configuring Scheduler Transmission Rate on page 284
- Allocation of Leftover Bandwidth on page 284

Example: Configuring Scheduler Transmission Rate

Configure the **best-effort** scheduler to use the remainder of the bandwidth on any interface to which it is assigned:

```
class-of-service {
  schedulers {
    best-effort {
      transmit-rate remainder;
    }
  }
}
```

Allocation of Leftover Bandwidth

The allocation of leftover bandwidth is a complex topic. It is difficult to predict and to test, because the behavior of the software varies depending on the traffic mix.

If a queue receives offered loads in excess of the queue's bandwidth allocation, the queue has negative bandwidth credit, and receives a share of any available leftover bandwidth. Negative bandwidth credit means the queue has used up its allocated bandwidth. If a queue's bandwidth credit is positive, meaning it is not receiving offered loads in excess of its bandwidth configuration, then the queue does not receive a share of leftover bandwidth. If the credit is positive, then the queue does not need to use leftover bandwidth, because it can use its own allocation.

Copyright © 2019, Juniper Networks, Inc.

This use of leftover bandwidth is the default. If you do not want a queue to use any leftover bandwidth, you must configure it for strict allocation by including the **transmit-rate** statement with the **exact** option at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level. With rate control in place, the specified bandwidth is strictly observed.

Juniper Networks M Series Multiservice Edge Routers and T Series Core Routers do not distribute leftover bandwidth in proportion to the configured transmit rate of the queues. Instead, the scheduler distributes the leftover bandwidth equally in round-robin fashion to queues that have negative bandwidth credit. All negative-credit queues can take the leftover bandwidth in equal share. This description suggests a simple round-robin distribution process among the queues with negative credits. In actual operation, a queue might change its bandwidth credit status from positive to negative and from negative to positive instantly while the leftover bandwidth is being distributed. Lower-rate queues tend to be allocated a larger share of leftover bandwidth, because their bandwidth credit is more likely to be negative at any given time, if they are overdriven persistently. Also, if there is a large packet size difference, (for example, queue 0 receives 64-byte packets, whereas queue 1 receives 1500-byte packets), then the actual leftover bandwidth distribution ratio can be skewed substantially, because each round-robin turn allows exactly one packet to be transmitted by a negative-credit queue, regardless of the packet size.

By default, on MX Series routers, the M320 Enhanced Type 4 FPCs, and T4000 routers with Type 5 FPCs and EX Series switches, excess bandwidth is shared in the ratio of the transmit rates. You can adjust this distribution by configuring the excess-rate statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level. You can specify the excess rate sharing by percentage or by proportion.

In summary, M Series and T Series routers distribute leftover bandwidth in equal shares for the queues with the same priority and same negative-credit status. MX Series routers and M320 Enhanced Type 4 FPCs, and EX Series switches, share excess bandwidth in the ratio of the transmit rates, but you can adjust this distribution.

Related Documentation

- Configuring Schedulers for Priority Scheduling on page 330
- How Schedulers Define Output Queue Properties on page 254
- Configuring a Scheduler on page 489
- excess-rate on page 1107
- schedulers on page 1262

Providing a Guaranteed Minimum Rate

On Gigabit Ethernet IQ PIC, EQ DPC, MIC, MPC, and Channelized IQ PIC interfaces, and on FRF.16 LSQ interfaces on Multiservices and Services PICs, you can configure guaranteed bandwidth, also known as a committed information rate (CIR). This allows you to specify a guaranteed rate for each logical interface. The guaranteed rate is a minimum. If excess physical interface bandwidth is available for use, the logical interface receives more than the guaranteed rate provisioned for the interface.

Copyright © 2019, Juniper Networks, Inc.

You cannot provision the sum of the guaranteed rates to be more than the physical interface bandwidth, or the bundle bandwidth for LSQ interfaces. If the sum of the guaranteed rates exceeds the interface or bundle bandwidth, the commit operation does not fail, but the software automatically decreases the rates so that the sum of the guaranteed rates is equal to the available bundle bandwidth.

To configure a guaranteed minimum rate, perform the following steps:

1. Include the **guaranteed-rate** statement at the **[edit class-of-service traffic-control-profile *profile-name*]** hierarchy level:

   ```
   [edit class-of-service traffic-control-profiles profile-name]
   guaranteed-rate (percent percentage | rate) <burst-size bytes>;
   ```

   On LSQ interfaces, you can configure the guaranteed rate as a percentage from 1 through 100.

   On IQ and IQ2 interfaces, you can configure the guaranteed rate as an absolute rate from 1000 through 6,400,000,000,000 bps.

   NOTE: For channelized and Gigabit Ethernet IQ interfaces, the **shaping-rate** and **guaranteed-rate** statements are mutually exclusive. You cannot configure some logical interfaces to use a shaping rate and others to use a guaranteed rate. This means there are no service guarantees when you configure a PIR. For these interfaces, you can configure either a PIR or a CIR, but not both.

   This restriction does not apply to Gigabit Ethernet IQ2 PICs or LSQ interfaces on Multiservices and Services PICs. For LSQ and Gigabit Ethernet IQ2 interfaces, you can configure both a PIR and a CIR on an interface.

   For more information about Gigabit Ethernet IQ2 PICs, see "CoS on Enhanced IQ2 PICs Overview" on page 794.

2. Optionally, you can base the delay-buffer calculation on a delay-buffer rate. To do this, include the **delay-buffer-rate** statement **[edit class-of-service traffic-control-profiles *profile-name*]** hierarchy level:

   ```
   [edit class-of-service traffic-control-profiles profile-name]
   delay-buffer-rate (percent percentage | rate);
   ```

   On LSQ interfaces, you can configure the delay-buffer rate as a percentage from 1 through 100.

   On IQ and IQ2 interfaces, you can configure the delay-buffer rate as an absolute rate from 1000 through 6,400,000,000,000 bps.

   The actual delay buffer is based on the calculations described in "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365. For an example showing how the delay-buffer rates are applied, see "Example: Providing a Guaranteed Minimum Rate" on page 289.

Copyright © 2019, Juniper Networks, Inc.

If you do not include the **delay-buffer-rate** statement, the delay-buffer calculation is based on the guaranteed rate, the shaping rate if no guaranteed rate is configured, or the scaled shaping rate if the interface is oversubscribed.

If you do not specify a shaping rate or a guaranteed rate, the logical interface receives a minimal delay-buffer rate and minimal bandwidth equal to four MTU-sized packets.

You can configure a rate for the delay buffer that is higher than the guaranteed rate. This can be useful when the traffic flow might not require much bandwidth in general, but in some cases traffic can be bursty and therefore needs a large buffer.

Configuring large buffers on relatively slow-speed links can cause packet aging. To help prevent this problem, the software requires that the sum of the delay-buffer rates be less than or equal to the port speed. This restriction does not eliminate the possibility of packet aging, so you should be cautious when using the **delay-buffer-rate** statement. Though some amount of extra buffering might be desirable for burst absorption, delay-buffer rates should not far exceed the service rate of the logical interface.

If you configure delay-buffer rates so that the sum exceeds the port speed, the configured delay-buffer rate is not implemented for the last logical interface that you configure. Instead, that logical interface receives a delay-buffer rate of 0, and a warning message is displayed in the CLI. If bandwidth becomes available (because another logical interface is deleted or deactivated, or the port speed is increased), the configured delay-buffer-rate is reevaluated and implemented if possible.

If the guaranteed rate of a logical interface cannot be implemented, that logical interface receives a delay-buffer rate of 0, even if the configured delay-buffer rate is within the interface speed. If at a later time the guaranteed rate of the logical interface can be met, the configured delay-buffer rate is reevaluated and if the delay-buffer rate is within the remaining bandwidth, it is implemented.

If any logical interface has a configured guaranteed rate, all other logical interfaces on that port that do not have a guaranteed rate configured receive a delay-buffer rate of 0. This is because the absence of a guaranteed rate configuration corresponds to a guaranteed rate of 0 and, consequently, a delay-buffer rate of 0.

3. To assign a scheduler map to the logical interface, include the **scheduler-map** statement at the **[edit class-of-service traffic-control-profiles *profile-name*]** hierarchy level:

   ```
   [edit class-of-service traffic-control-profiles profile-name]
   scheduler-map map-name;
   ```

   For information about configuring schedulers and scheduler maps, see "Configuring Schedulers" on page 258 and "Configuring Scheduler Maps" on page 258.

4. To enable large buffer sizes to be configured, include the **q-pic-large-buffer** statement at the **[edit chassis fpc *slot-number* pic *pic-number*]** hierarchy level:

   ```
   [edit chassis fpc slot-number pic pic-number]
   q-pic-large-buffer;
   ```

Copyright © 2019, Juniper Networks, Inc.

If you do not include this statement, the delay-buffer size is more restricted. For more information, see "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365.

5. To enable scheduling on logical interfaces, include the **per-unit-scheduler** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
per-unit-scheduler;
```

When you include this statement, the maximum number of VLANs supported is 768 on a single-port Gigabit Ethernet IQ PIC. On a dual-port Gigabit Ethernet IQ PIC, the maximum number is 384.

6. To apply the traffic-scheduling profile to the logical interface, include the output-traffic-control-profile statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
output-traffic-control-profile profile-name;
```

Table 29 on page 288 shows how the bandwidth and delay buffer are allocated in various configurations.

Table 29: Bandwidth and Delay Buffer Allocations by Configuration Scenario

Configuration Scenario	Delay Buffer Allocation
You do not configure a guaranteed rate. You do not configure a delay-buffer rate.	Logical interface receives minimal bandwidth with no guarantees and receives a minimal delay buffer equal to 4 MTU-sized packets.
You configure a guaranteed rate. You do not configure a delay-buffer rate.	Logical interface receives bandwidth equal to the guaranteed rate and a delay buffer based on the guaranteed rate. The multiplicative factor depends on whether you include the **q-pic-large-buffer** statement. For more information, see "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365.
You configure a guaranteed rate. You configure a delay-buffer rate.	Logical interface receives bandwidth equal to the guaranteed rate and a delay buffer based on the delay-buffer rate. The multiplicative factor depends on whether you include the **q-pic-large-buffer** statement. For more information, see "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365.

Verifying Configuration of Guaranteed Minimum Rate

To verify your configuration, you can issue this following operational mode commands:

- **show class-of-service interfaces**
- **show class-of-service traffic-control-profile *profile-name***

Copyright © 2019, Juniper Networks, Inc.

Example: Providing a Guaranteed Minimum Rate

Two logical interface units, **0** and **1**, are provisioned with a guaranteed minimum of 750 Kbps and 500 Kbps, respectively. For logical unit **1**, the delay buffer is based on the guaranteed rate setting. For logical unit **0**, a delay-buffer rate of 500 Kbps is specified. The actual delay buffers allocated to each logical interface are 2 seconds of 500 Kbps. The 2-second value is based on the following calculation:

```
delay-buffer-rate < [8 x 64 Kbps]): 2 seconds of delay-buffer-rate
```

For more information about this calculation, see "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365.

```
chassis {
  fpc 3 {
    pic 0 {
      q-pic-large-buffer;
    }
  }
}
interfaces {
  t1-3/0/1 {
    per-unit-scheduler;
  }
}
class-of-service {
  traffic-control-profiles {
    tc-profile3 {
      guaranteed-rate 750k;
      scheduler-map sched-map3;
      delay-buffer-rate 500k; # 500 Kbps is less than 8 x 64 Kbps
    }
    tc-profile4 {
      guaranteed-rate 500k; # 500 Kbps is less than 8 x 64 Kbps
      scheduler-map sched-map4;
    }
  }
  interfaces {
    t1-3/0/1 {
    unit 0 {
      output-traffic-control-profile tc-profile3;
    }
    unit 1 {
      output-traffic-control-profile tc-profile4;
    }
  }
}
```

Copyright © 2019, Juniper Networks, Inc.

PIR-Only and CIR Mode

The actual behavior of many CoS parameters, especially the shaping rate and guaranteed rate, depend on whether the physical interface is operating in PIR-only or CIR mode.

In PIR-only mode, one or more nodes perform shaping. The physical interface is in the PIR-only mode if no child (or grandchild) node under the port has a guaranteed rate configured.

The mode of the port is important because in PIR-only mode, the scheduling across the child nodes is in proportion to their shaping rates (PIRs) and not the guaranteed rates (CIRs). This can be important if the observed behavior is not what is anticipated.

In CIR mode, one or more nodes applies a guaranteed rate and might perform shaping. A physical interface is in CIR mode if at least one child (or grandchild) node has a guaranteed rate configured.

In CIR mode, one or more nodes applies the guaranteed rates. In addition, any child or grandchild node under the physical interface can have a shaping rate configured. Only the guaranteed rate matters. In CIR mode, nodes that do not have a guaranteed rate configured are assumed to have a very small guaranteed rate (queuing weight).

Excess Rate and Excess Priority Configuration Examples

To configure the excess rate for nonqueuing Packet Forwarding Engines, include the excess-rate statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level.

To configure the excess priority for nonqueuing Packet Forwarding Engines, include the excess-priority statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level.

The relationship between the configured guaranteed rate, excess rate, guaranteed priority, excess priority, and offered load is not always obvious. The following tables show the expected throughput of a Gigabit Ethernet port with various bandwidth-sharing parameters configured on the queues.

The default behavior of a nonqueuing Gigabit Ethernet interface with multiple priority levels is shown in Table 30 on page 290. All queues in the table get their guaranteed rate. The excess bandwidth is first offered to the excess high-priority queues. Because these use all available bandwidth, these is no remaining excess bandwidth for the low-priority queues.

Table 30: Current Behavior with Multiple Priority Levels

Queue	Guaranteed (Transmit) Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q0	20%	high	high	600 Mbps	200 + 366.67 = 566.67 Mbps
Q1	10%	high	high	500 Mbps	100 + 183.33 = 283.33 Mbps

Copyright © 2019, Juniper Networks, Inc.

Table 30: Current Behavior with Multiple Priority Levels (continued)

Queue	Guaranteed (Transmit) Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q2	10%	low	low	500 Mbps	100 + 0 = 100 Mbps
Q3	5%	low	low	500 Mbps	50 + 0 = 50 Mbps

The default behavior of a nonqueuing Gigabit Ethernet interface with the same priority levels is shown in Table 31 on page 291. All queues in the table get their guaranteed rate. Because all queues have the same excess priority, they share the excess bandwidth and each queue gets excess bandwidth in proportion to the transmit rate.

Table 31: Current Behavior with Same Priority Levels

Queue	Guaranteed (Transmit) Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q0	20%	high	high	500 Mbps	200 + 244.44 = 444.44 Mbps
Q1	10%	high	high	500 Mbps	100 + 122.22 = 222.22 Mbps
Q2	10%	high	high	500 Mbps	100 + 122.22 = 222.22 Mbps
Q3	5%	high	high	500 Mbps	50 + 61.11= 111.11 Mbps

The default behavior of a nonqueuing Gigabit Ethernet interface with the at least one strict-high priority level is shown in Table 32 on page 291. First the high priority and strict-high are serviced in a weighted round-robin fashion. The high priority queue gets its guaranteed bandwidth and the strict-high queue gets what remains. The high excess priority queue gets all the excess bandwidth.

Table 32: Current Behavior with Strict-High Priority

Queue	Guaranteed (Transmit) Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q0	20%	strict-high	X	500 Mbps	500 Mbps
Q1	10%	high	high	500 Mbps	100 + 250 = 350 Mbps
Q2	10%	low	low	500 Mbps	100 + 0 = 100 Mbps
Q3	5%	low	low	500 Mbps	50 + 0= 50 Mbps

The default behavior of a nonqueuing Gigabit Ethernet interface with the at least one strict-high priority level and a higher offered load on Q0 is shown in Table 33 on page 292. First the high priority and strict-high are serviced in a weighted round-robin fashion. The high priority queue gets its guaranteed bandwidth and the strict-high queue gets what remains. (The high priority queue receives its guaranteed bandwidth unless a strict-high

Copyright © 2019, Juniper Networks, Inc.

queue is configured, which in certain conditions might starve the high priority queue. To guarantee the configured transmit rate on high-priority queues, apply the **rate-limit** option to the transmit rate of the strict-high priority queue.) There is no excess bandwidth.

Table 33: Strict-High Priority with Higher Load

Queue	Guaranteed (Transmit) Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q0	20%	strict-high	X	1 Gbps	900 Mbps
Q1	10%	high	high	500 Mbps	100 + 0 = 100 Mbps
Q2	10%	low	low	500 Mbps	0 + 0 = 0 Mbps
Q3	5%	low	low	500 Mbps	0 + 0= 0 Mbps

Now consider the behavior of the queues with configured excess rates and excess priorities.

The behavior with multiple priority levels is shown in Table 34 on page 292. All queues get the guaranteed rate. The excess bandwidth is first offered to the excess high priority queues and these consume all the bandwidth. There is no remaining excess bandwidth for low priority queues.

Table 34: Sharing with Multiple Priority Levels

Queue	Guaranteed (Transmit) Rate	Excess Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q0	20%	10%	high	high	500 Mbps	200 + 275 = 475 Mbps
Q1	10%	20%	high	low	500 Mbps	100 + 0 = 100 Mbps
Q2	10%	10%	low	high	500 Mbps	100 + 275 = 275 Mbps
Q3	5%	20%	low	low	500 Mbps	50 + 0= 50 Mbps

The behavior with the same (high) priority levels is shown in Table 35 on page 292. All queues get the guaranteed rate. Because all queues have the same excess priority, they share the excess bandwidth in proportion to their transmit rate.

Table 35: Sharing with the Same Priority Levels

Queue	Guaranteed (Transmit) Rate	Excess Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q0	20%	10%	high	high	500 Mbps	200 + 91.67 = 291.67 Mbps
Q1	10%	20%	high	high	500 Mbps	100 + 183.33 = 283.33 Mbps

Copyright © 2019, Juniper Networks, Inc.

Table 35: Sharing with the Same Priority Levels (continued)

Queue	Guaranteed (Transmit) Rate	Excess Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q2	10%	10%	high	high	500 Mbps	100 + 91.67 = 191.67 Mbps
Q3	5%	20%	high	high	500 Mbps	50 + 183.33 = 233.33 Mbps

The behavior with at least one strict-high priority level is shown in Table 36 on page 293. The high priority and strict-high queues are serviced in a weighted round-robin fashion. The high priority queue gets its guaranteed rate and the strict-high queue gets the rest. The excess high-priority queue get all the excess bandwidth.

Table 36: Sharing with at Least One Strict-High Priority

Queue	Guaranteed (Transmit) Rate	Excess Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q0	20%	X	strict-high	X	500 Mbps	500 Mbps
Q1	10%	20%	high	low	500 Mbps	100 + 0 = 100 Mbps
Q2	10%	10%	low	high	500 Mbps	100 + 250 = 350 Mbps
Q3	5%	20%	low	low	500 Mbps	50 + 0 = 50 Mbps

The behavior with at least one strict-high priority level and a higher offered load is shown in Table 37 on page 293. The high priority and strict-high queues are serviced in a weighted round-robin fashion. The high priority queue gets its guaranteed rate and the strict-high queue gets the rest. There is no excess bandwidth.

Table 37: Sharing with at Least One Strict-High Priority and Higher Load

Queue	Guaranteed (Transmit) Rate	Excess Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q0	20%	X	strict-high	X	900 Mbps	900 Mbps
Q1	10%	20%	high	low	500 Mbps	100 + 0 = 100 Mbps
Q2	10%	10%	low	high	500 Mbps	0 + 0 = 0 Mbps
Q3	5%	20%	low	low	500 Mbps	0 + 0 = 0 Mbps

The behavior with at least one strict-high priority level and a rate limit is shown in Table 38 on page 294. Queue 0 and Queue 2 are rate limited, so the maximum bandwidth they are offered is the transmit bandwidth and they will not be offered any excess

Copyright © 2019, Juniper Networks, Inc.

bandwidth. All other queues are offered the guaranteed bandwidth and the excess is shared by the non-rate-limited queues.

Table 38: Sharing with at Least One Strict-High Priority and Rate Limit

Queue	Guaranteed (Transmit) Rate	Rate Limit	Excess Rate	Guaranteed Priority	Excess Priority	Offered Load	Expected Throughput
Q0	20%	Yes	X	strict-high	X	500 Mbps	200 + 0 = 200 Mbps
Q1	10%	No	20%	high	low	500 Mbps	100 + 275 = 375 Mbps
Q2	10%	Yes	10%	low	high	500 Mbps	100 + 0 = 100 Mbps
Q3	5%	No	20%	low	low	500 Mbps	50 + 275 = 325 Mbps

Configuring the Schedulers

The following example configures schedulers, forwarding classes, and a scheduler map for an interface with excess rates and excess priorities.

```
[edit class-of-service schedulers]
scheduler-1 {
   transmit-rate percent 20;
   priority high;
   excess-rate percent 10;
   excess-priority low;
}
scheduler-2 {
   transmit-rate percent 10;
   priority strict-high;
}
scheduler-3 {
   transmit-rate percent 10;
   priority medium-high;
   excess-rate percent 20;
   excess-priority high;
}
scheduler-4 {
   transmit-rate percent 5;
   priority medium-high;
   excess-rate percent 30;
   excess-priority low;
}
```

Configuring the Forwarding Classes

```
[edit class-of-service]
forwarding-classes {
   class cp_000 queue-num 0;
   class cp_001 queue-num 1;
```


Copyright © 2019, Juniper Networks, Inc.

```
    class cp_010 queue-num 2;
    class cp_011 queue-num 3;
    class cp_100 queue-num 4;
    class cp_101 queue-num 5;
    class cp_110 queue-num 6;
    class cp_111 queue-num 7;
}
```

Configuring the Scheduler Map

```
[edit class-of-service scheduler-maps]
scheduler-map-1 {
    forwarding-class cp_000 scheduler scheduler-1;
    forwarding-class cp_001 scheduler scheduler-2;
    forwarding-class cp_010 scheduler scheduler-3;
    forwarding-class cp_011 scheduler scheduler-4;
}
```

Applying the Scheduler Map to the Interface

```
[edit interfaces]
ge-1/1/0 {
    scheduler-map scheduler-map-1;
    unit 0 {
        family inet {
            address 192.168.1.2/32;
        }
    }
}
```

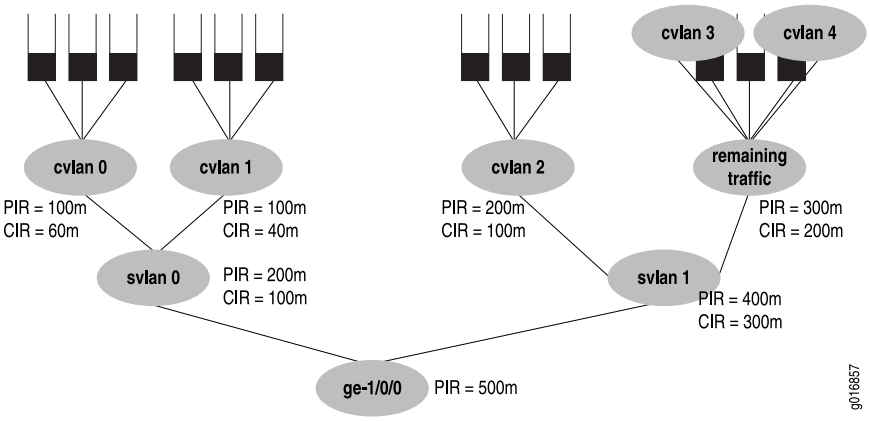
Controlling Remaining Traffic

You can configure many logical interfaces under an interface. However, only a subset of them might have a traffic control profile attached. For example, you can configure three logical interfaces (units) over the same service VLAN, but apply a traffic control profile specifying best-effort and voice queues to only one of the logical interface units. Traffic from the two remaining logical interfaces is considered *remaining traffic*. To configure transmit rate guarantees for the remaining traffic, you configure the **output-traffic-control-profile-remaining** statement specifying a guaranteed rate for the remaining traffic. Without this statement, the remaining traffic gets a default, minimal bandwidth. In the same way, the **shaping-rate** and **delay-buffer-rate** statements can be specified in the traffic control profile referenced with the **output-traffic-control-profile-remaining** statement in order to shape and provide buffering for remaining traffic.

Consider the interface shown in Figure 32 on page 296. Customer VLANs 3 and 4 have no explicit traffic control profile. However, the service provider might want to establish a shaping and guaranteed transmit rate for aggregate traffic heading for those customer VLANs. The solution in to configure and apply a traffic control profile for all remaining traffic on the interface.

Copyright © 2019, Juniper Networks, Inc.

Figure 32: Handling Remaining Traffic



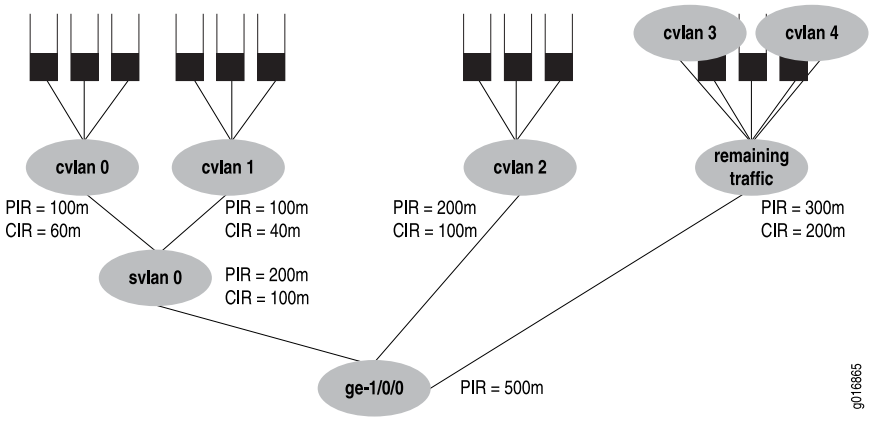
This example considers the case where customer VLANs 3 and 4 have no explicit traffic control profile, yet need to establish a shaping and guaranteed transmit rate for traffic heading for those customer VLANs. The solution is to add a traffic control profile to the **svlan1** interface set. This example builds on the earlier example and so does not repeat all configuration details, only those at the service VLAN level.

```
[edit class-of-service interfaces]
interface-set svlan0 {
   output-traffic-control-profile tcp-svlan0;
}
interface-set svlan1 {
   output-traffic-control-profile tcp-svlan1; # For explicitly shaped traffic.
   output-traffic-control-profile-remaining tcp-svlan1-remaining; # For all remaining traffic.
}

[edit class-of-service traffic-control-profiles]
tcp-svlan1 {
   shaping-rate 400m;
   guaranteed-rate 300m;
}
tcp-svlan1-remaining {
   shaping-rate 300m;
   guaranteed-rate 200m;
   scheduler-map smap-remainder; # this smap is not shown in detail
}
```

Next, consider the example shown in Figure 33 on page 297.

Copyright © 2019, Juniper Networks, Inc.

Figure 33: Another Example of Handling Remaining Traffic



In this example, **ge-1/0/0** has three logical interfaces (unit 1, unit 2, and unit 3), and SVLAN 2000, which are covered by the interface set:

- Scheduling for the interface set **svlan0** is specified by referencing an **output-traffic-control-profile** statement which specifies the **guaranteed-rate**, **shaping-rate**, and **delay-buffer-rate** statement values for the interface set. In this example, the output traffic control profile called **tcp-svlan0** guarantees 100 Mbps and shapes the interface set **svlan0** to 200 Mbps.
- Scheduling and queuing for remaining traffic of **svlan0** is specified by referencing an **output-traffic-control-profile-remaining** statement which references a **scheduler-map** statement that establishes queues for the remaining traffic. The specified traffic control profile can also configure guaranteed, shaping, and delay-buffer rates for the remaining traffic. In this example, **output-traffic-control-profile-remaining tcp-svlan0-rem** references **scheduler-map smap-svlan0-rem**, which calls for a best-effort queue for remaining traffic (that is, traffic on unit 3 and unit 4, which is not classified by the **svlan0** interface set). The example also specifies a **guaranteed-rate** of 200 Mbps and a **shaping-rate** of 300 Mbps for all remaining traffic.
- Scheduling and queuing for logical interface **ge-1/0/0 unit 1** is configured "traditionally" and uses an **output-traffic-control-profile** specified for that unit. In this example, **output-traffic-control-profile tcp-ifl1** specifies scheduling and queuing for **ge-1/0/0 unit 1**.

This example does not include the **[edit interfaces]** configuration.

```
[edit class-of-service interfaces]
interface-set {
   svlan0 {
      output-traffic-control-profile tcp-svlan0; # Guarantee & shaper for svlan0.
   }
}
ge-1/0/0 {
   output-traffic-control-profile-remaining tcp-svlan0-rem;
   # Unit 3 and 4 are not explicitly configured, but captured by "remaining"
   unit 1 {
      output-traffic-control-profile tcp-ifl1; # Unit 1 be & ef queues.
   }
```

Copyright © 2019, Juniper Networks, Inc.

```
}
```

Here is how the traffic control profiles for this example are configured:

```
[edit class-of-service traffic-control-profiles]
tcp-svlan0 {
   shaping-rate 200m;
   guaranteed-rate 100m;
}
tcp-svlan0-rem {
   shaping-rate 300m;
   guaranteed-rate 200m;
   scheduler-map smap-svlan0-rem; # This specifies queues for remaining traffic
}
tcp-ifl1 {
   scheduler-map smap-ifl1;
}
```

Finally, here are the scheduler maps and queues for the example:

```
[edit class-of-service scheduler-maps]
smap-svlan0-rem {
   forwarding-class best-effort scheduler sched-foo;
}
smap-ifl1 {
   forwarding-class best-effort scheduler sched-bar;
   forwarding-class assured-forwarding scheduler sched-baz;
}
```

The configuration for the referenced schedulers are not given for this example.

Bandwidth Sharing on Nonqueuing Packet Forwarding Engines Overview

You can configure bandwidth sharing rate limits, excess rate, and excess priority at the queue level on the following Juniper Networks routers and switches:

- EX Series switches
- M120 Multiservice Edge Router (rate limit and excess priority only; excess rate is not configured by the user)
- M320 router with Enhanced FPCs (rate limit, excess rate, and excess priority)
- MX Series 5G Universal Routing Platform with nonqueuing DPCs (rate limit, excess rate, and excess priority)

You configure rate limits when you have a concern that low-latency packets (such as high or strict-high priority packets for voice) might starve low-priority and medium-priority packets. In Junos OS, the low latency queue is implemented by rate-limiting packets to the transmit bandwidth. The rate-limiting is performed immediately before queuing the packet for transmission. All packets that exceed the rate limit are not queued, but dropped.

By default, if the excess priority is not configured for a queue, the excess priority will be the same as the normal queue priority. If none of the queues have an excess rate

Copyright © 2019, Juniper Networks, Inc.

configured, then the excess rate will be the same as the transmit rate percentage. If at least one of the queues has an excess rate configured, then the excess rate for the queues that do not have an excess rate configured will be set to zero.

When the physical interface is on queuing hardware such as the IQ, IQ2, or IQE PICs, or MX Series routers queuing DPCs or EX Series switches, these features are dependent on the PIC (or queuing DPC in the case of the MX Series router) configuration.

You cannot configure both rate limits and buffer sizes on these Packet Forwarding Engines.

Four levels of excess priorities are supported: low, medium-low, medium-high, and high.

NOTE: Rate limiting is implemented differently on Enhanced Queuing DPCs and non-queuing Packet Forwarding Engines. On Enhanced Queuing DPCs, rate-limiting is implemented using a single rate two color policer. On non-queuing Packet Forwarding Engines, rate-limiting is achieved by shaping the queue to the transmit rate and keeping the queue delay buffers small to prevent too many packets from being queued once the shaping rate is reached.

Configuring Rate Limits on Nonqueuing Packet Forwarding Engines

On non-queuing Packet Forwarding Engines, rate-limiting is achieved by shaping the queue to the transmit rate and keeping the queue delay buffers small to prevent too many packets from being queued once the shaping rate is reached. To configure rate limits for non-queuing Packet Forwarding Engines, include the **transmit-rate** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level.

NOTE: Rate limiting is implemented differently on MPCs and Enhanced Queuing DPCs than on non-queuing Packet Forwarding Engines. On MPCs and Enhanced Queuing DPCs, rate-limiting is implemented using a single-rate two-color policer. See "Example: Limiting Outbound Traffic Within Your Network by Configuring an Egress Single-Rate Two-Color Policer and Configuring Multifield Classifiers" on page 155 for an example of configuring a single-rate two-color policer to rate limit traffic.

Configuring the Schedulers

The following example configures schedulers, forwarding classes, and a scheduler map for a rate-limited interface.

```
[edit class-of-service schedulers]
scheduler-1 {
   transmit-rate percent 20 rate-limit;
   priority high;
}
scheduler-2 {
   transmit-rate percent 10 rate-limit;
   priority strict-high;
```

Copyright © 2019, Juniper Networks, Inc.

```
}
scheduler-3 {
   transmit-rate percent 40;
   priority medium-high;
}
scheduler-4 {
   transmit-rate percent 30;
   priority medium-high;
}
```

Configuring the Forwarding Classes

```
[edit class-of-service]
forwarding-classes {
   class cp_000 queue-num 0;
   class cp_001 queue-num 1;
   class cp_010 queue-num 2;
   class cp_011 queue-num 3;
   class cp_100 queue-num 4;
   class cp_101 queue-num 5;
   class cp_110 queue-num 6;
   class cp_111 queue-num 7;
}
```

Configuring the Scheduler Map

```
[edit class-of-service scheduler-maps]
scheduler-map-1 {
   forwarding-class cp_000 scheduler scheduler-1;
   forwarding-class cp_001 scheduler scheduler-2;
   forwarding-class cp_010 scheduler scheduler-3;
   forwarding-class cp_011 scheduler scheduler-4;
}
```

Applying the Scheduler Map to the Interface

```
[edit class-of-service interfaces]
ge-1/0/0 {
   scheduler-map scheduler-map-1;
}
```

Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs

By default, output scheduling is not enabled on logical interfaces. Logical interfaces without shaping configured share a default scheduler. This scheduler has a committed information rate (CIR) that equals 0. (The CIR is the guaranteed rate.) The default scheduler has a peak information rate (PIR) that equals the physical interface shaping rate.

Copyright © 2019, Juniper Networks, Inc.

NOTE: If you apply a shaping rate, you must keep in mind that the transit statistics for physical interfaces are obtained from the packet forwarding engine, but the traffic statistics are supplied by the PIC. Therefore, if shaping is applied to the PIC, the count of packets in the transit statistics fields do not always agree with the counts in the traffic statistics. For example, the IPv6 transit statistics will not necessarily match the traffic statistics on the interface. However, at the logical interface (DLCI) level, both transit and traffic statistics are obtained from the Packet Forwarding Engine and will not show any difference.

Logical interface scheduling (also called *per-unit scheduling*) allows you to enable multiple output queues on a logical interface and associate customized output scheduling and shaping for each queue.

NOTE: Ingress scheduling does not support logical interface scheduling.

You can configure logical interface scheduling on the following PICs:

- Multiservices and Services PICs , on link services IQ (**lsq-**) interfaces
- Channelized E1 IQ PIC
- Channelized OC3 IQ PIC
- Channelized OC12 IQ PIC (Per-unit scheduling is not supported on T1 interfaces configured on this PIC.)
- Channelized STM1 IQ PIC
- Channelized T3 IQ PIC
- E3 IQ PIC
- Gigabit Ethernet IQ PIC
- Gigabit Ethernet IQ2 PIC
- IQE PICs

You can configure logical interface scheduling on the following MICs and MPCs as well as any MPC that contains a queuing chip:

- 16x10GE MPC
- MPC3E:
  - 2x10GE MIC with XFP
  - 10x10GE MIC with SFP+
  - 2x40GE MIC with QSFP+
  - 1x100GE MIC with CXP

Copyright © 2019, Juniper Networks, Inc.

- MPC4E:
  - 32x10GE with SFPP
  - 2x100GE + 8x10GE with SFPP
- MPC6E:
  - 24x10GE MIC with SFPP
  - 24x10GE MIC with SFPP OTN
  - 2x100GE MIC with CFP2 OTN
  - 4x100GE MIC with CXP

For Channelized and Gigabit Ethernet IQ PICs only, you can configure a shaping rate for a VLAN or DLCI and oversubscribe the physical interface by including the **shaping-rate** statement at the **[edit class-of-service traffic-control-profiles]** hierarchy level. With this configuration approach, you can independently control the delay-buffer rate, as described in "Oversubscribing Interface Bandwidth" on page 273.

Physical interfaces (for example, **t3-0/0/0**, **t3-0/0/0:0**, and **ge-0/0/0**) support scheduling with any encapsulation type pertinent to that physical interface. For a single port, you cannot apply scheduling to the physical interface if you apply scheduling to one or more of the associated logical interfaces.

For Gigabit Ethernet IQ2 PIC PICs only, you can configure hierarchical traffic shaping, meaning the shaping is performed on both the physical interface and the logical interface. You can also configure input traffic scheduling and shared scheduling. For more information, see "CoS on Enhanced IQ2 PICs Overview" on page 794.

Logical interfaces (for example. **t3-0/0/0.0**, **ge-0/0/0.0**, and **t1-0/0/0:0.1**) support scheduling on DLCIs or VLANs only. Furthermore, logical interface scheduling is not supported on PICs that do not have IQ.

Copyright © 2019, Juniper Networks, Inc.

NOTE: In the Junos OS implementation, the term *logical interfaces* generally refers to interfaces you configure by including the **unit** statement at the **[edit interfaces *interface-name*]** hierarchy level. As such, logical interfaces have the *logical* descriptor at the end of the interface name, as in **ge-0/0/0.1** or **t1-0/0/0:0.1**, where the logical unit number is **1**.

Although channelized interfaces are generally thought of as logical or virtual, the Junos OS sees T3, T1, and *N*xDS0 interfaces within a channelized IQ PIC as physical interfaces. For example, both **t3-0/0/0** and **t3-0/0/0:1** are treated as physical interfaces by the Junos OS. In contrast, **t3-0/0/0.2** and **t3-0/0/0:1.2** are considered logical interfaces because they have the **.2** at the end of the interface names.

Within the **[edit class-of-service]** hierarchy level, you cannot use the ***.logical*** descriptor when you assign properties to logical interfaces. Instead, you must include the **unit** statement in the configuration. For example:

```
[edit class-of-service]
user@host# set interfaces t3-0/0/0 unit 0 scheduler-map map1
```

Table 39 on page 303 shows the interfaces/PICs that support fine-grained queuing and scheduling.

Table 39: Fine-Grained Queuing and Scheduling Support by Interface or PIC Type

Interface Type	PIC Type	Supported	Example Configuration
IQ PICs			
Physical interfaces	ATM2 IQ	Yes	Example of supported configuration: **[edit class-of-service interfaces at-0/0/0]** **scheduler-map map-1;**
Channelized interfaces configured on IQ PICs	Channelized DS3 IQ	Yes	Example of supported configuration: **[edit class-of-service interfaces t1-0/0/0:1]** **scheduler-map map-1;**

Copyright © 2019, Juniper Networks, Inc.

Table 39: Fine-Grained Queuing and Scheduling Support by Interface or PIC Type (continued)

Interface Type	PIC Type	Supported	Example Configuration
Logical interfaces (DLCIs and VLANs only) configured on IQ PICs	Gigabit Ethernet IQ with VLAN tagging enabled	Yes	Example of supported configuration: **[edit class-of-service interfaces ge-0/0/0 unit 1]** **scheduler-map map-1;**
	E3 IQ with Frame Relay encapsulation	Yes	Example of supported configuration: **[edit class-of-service interfaces e3-0/0/0 unit 1]** **scheduler-map map-1;**
	Channelized OC3 IQ with Frame Relay encapsulation	Yes	Example of supported configuration: **[edit class-of-service interfaces t1-1/0/0:1:1 unit 0]** **scheduler-map map-1;**
	Channelized STM1 IQ with Frame Relay encapsulation	Yes	Example of supported configuration: **[edit class-of-service interfaces e1-0/0/0:1 unit 1]** **scheduler-map map-1;**
	Channelized T3 IQ with Frame Relay encapsulation	Yes	Example of supported configuration: **[edit class-of-service interfaces t1-0/0/0 unit 1]** **scheduler-map map-1;**
Logical interfaces configured on IQ PICs (interfaces that are not DLCIs or VLANs)	E3 IQ PIC with Cisco HDLC encapsulation	No	No
	ATM2 IQ PIC with LLC/SNAP encapsulation	No	No
	Channelized OC12 IQ PIC with PPP encapsulation	No	No
Non-IQ PICs			
Physical interfaces	T3	Yes	Example of supported configuration: **[edit class-of-service interfaces t3-0/0/0]** **scheduler-map map-1;**
Channelized OC12 PIC	Channelized OC12	Yes	Example of supported configuration: **[edit class-of-service interfaces t3-0/0/0:1]** **scheduler-map map-1;**
Channelized interfaces (except the Channelized OC12 PIC)	Channelized STM1	No	No

Copyright © 2019, Juniper Networks, Inc.

Table 39: Fine-Grained Queuing and Scheduling Support by Interface or PIC Type (continued)

Interface Type	PIC Type	Supported	Example Configuration
Logical interfaces	Fast Ethernet	No	No
	Gigabit Ethernet	No	No
	ATM1	No	No
	Channelized OC12	No	No

Table 40 on page 305 shows the MICs and MPCs that support fine-grained queuing and scheduling.

Table 40: Fine-Grained Queuing and Scheduling Support by MIC or MPC Type

MPC	MIC	Supported	Example Configuration
Fixed Configuration MPCs			
16x10GE MPC	No	Yes	[edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
32x10GE MPC4E	No	Yes	[edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
2x100GE + 8x10GE MPC4E	No	Yes	[edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
6x40GE + 24x10GE MPC5E	No	No	No
6x40GE + 24x10GE MPC5EQ	No	Yes	[edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
2x100GE + 4x10GE MPC5E	No	No	No
2x100GE + 4x10GE MPC5EQ	No	Yes	[edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
MPCs			
MPC1	No	No	No
MPC1E	No	No	No
MPC1 Q	Any supported MIC	Yes	Example of supported configuration: [edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;

Copyright © 2019, Juniper Networks, Inc.

Table 40: Fine-Grained Queuing and Scheduling Support by MIC or MPC Type (continued)

MPC	MIC	Supported	Example Configuration
MPC1E Q	Any supported MIC	Yes	Example of supported configuration: [edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
MPC2	No	No	No
MPC2E	No	No	No
MPC2 Q	Any supported MIC	Yes	Example of supported configuration: [edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
MPC2E Q	Any supported MIC	Yes	Example of supported configuration: [edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
MPC2 EQ	Any supported MIC	Yes	Example of supported configuration: [edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
MPC2E EQ	Any supported MIC	Yes	Example of supported configuration: [edit class-of-service interfaces ge-0/0/0 unit 1] scheduler-map map-1;
MPC2E P	No	No	No
MPC3E	10-Gigabit Ethernet MIC with SFP+	Yes	Example of supported configuration: [edit class-of-service interfaces xe-0/0/0 unit 1] scheduler-map map-1;
	40-Gigabit Ethernet MIC with QSFP+	Yes	Example of supported configuration: [edit class-of-service interfaces et-0/0/0 unit 1] scheduler-map map-1;
	100-Gigabit Ethernet MIC with CXP	Yes	Example of supported configuration: [edit class-of-service interfaces et-0/0/0 unit 1] scheduler-map map-1;
MPC6E	Any supported MIC	Yes	Example of supported configuration: [edit class-of-service interfaces et-0/0/0 unit 1] scheduler-map map-1;

To configure scheduling on logical interfaces:

Copyright © 2019, Juniper Networks, Inc.

1. Enable per-unit scheduling on the interface by including the **per-unit-scheduler** statement at the **[edit interfaces *interface-name*]** hierarchy level:

   ```
   [edit interfaces interface-name]
   per-unit-scheduler;
   ```

   When including the **per-unit-scheduler** statement, you must also include the **vlan-tagging** statement or the **flexible-vlan-tagging** statement (to apply scheduling to VLANs) or the **encapsulation frame-relay** statement (to apply scheduling to DLCIs) at the **[edit interfaces *interface-name*]** hierarchy level.

   When you include this statement, the maximum number of VLANs supported is 768 on a single-port Gigabit Ethernet IQ PIC. On a dual-port Gigabit Ethernet IQ PIC, the maximum number is 384.

   See "Scaling of Per-VLAN Queuing on Non-Queuing MPCs" on page 935 for scaling information on non-queuing MPCs.

2. Associate a scheduler with the interface by including the **scheduler-map** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

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

   Alternatively, associate a scheduler with the interface by including the **scheduler-map** statement at the **[edit class-of-service traffic-control-profiles *traffic control profile name*]** hierarchy level and then include the **output-traffic-control-profile** statement at the **[edit class-of-service interfaces *interface name* unit *logical unit number*]** hierarchy level.

   ```
   [edit class-of-service traffic-control-profiles traffic control profile name]
   scheduler-map map-name;
   ```

   ```
   [edit class-of-service interfaces interface-name unit logical-unit-number]
   output-traffic-control-profile traffic-control-profile-name;
   ```

3. Configure shaping on the interface by including the **shaping-rate** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

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

   NOTE: You can also apply the shaping rate to the traffic control profile.

   By default, the logical interface bandwidth is the average of unused bandwidth for the number of logical interfaces that require default bandwidth treatment. You can specify a peak bandwidth rate in bps, 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). The range is from 1000 through 6,400,000,000,000 bps. For

Copyright © 2019, Juniper Networks, Inc.

the IQ2 Gigabit Ethernet PIC, the minimum is 80,000 bps, and for the IQ2 10 Gigabit Ethernet PIC, the minimum is 160,000 bps. For the 16x10GE MPC, the minimum is 250,000 bps, and for the MPC3E, MPC4E, and MPC6E, the minimum is 292,000 bps.

For FRF.16 bundles on link services interfaces, only shaping rates based on percentage are supported.

Related Documentation

- per-unit-scheduler on page 1226
- Example: Applying Scheduling and Shaping to VLANs on page 312
- Example: Applying Scheduler Maps and Shaping Rate to DLCIs on page 308

Example: Applying Scheduler Maps and Shaping Rate to DLCIs

This example shows how to apply scheduler maps and shaping rates to individual logical interfaces.

- Requirements on page 308
- Overview on page 308
- Configuration on page 309

Requirements

This example uses the following hardware and software components:

- Junos OS Release 7.4 or later running on router line cards that support Intelligent Queuing (IQ).
- Junos OS Release 13.2 or later running on MX Series routers containing 16x10GE MPC or MPC3E line cards.
- Junos OS Release 13.3 or later running on MX Series routers containing MPC4E line cards.
- Junos OS Release 15.1 or later running on MX Series routers containing MPC6E line cards.

Overview

By default, output scheduling is not enabled on logical interfaces. Logical interfaces without shaping configured share a default scheduler. *Logical interface scheduling* (also called *per-unit scheduling*) allows you to enable multiple output queues on a logical interface and associate customized scheduling and shaping for each queue.

This example shows how to define schedulers for logical interfaces through the direct use of scheduler maps and shaping rates.

In this example, we associate the scheduler **sched-map-logical-0** with logical interface **unit 0** on physical interface **t3-1/0/0**, and allocate 10 Mbps of transmission bandwidth to the logical interface. We also associate the scheduler **sched-map-logical-1** with logical

Copyright © 2019, Juniper Networks, Inc.

interface **unit 1** on the same physical interface, **t3-1/0/0**, and allocate 20 Mbps of transmission bandwidth to the logical interface.

The allocated bandwidth is shared among the individual forwarding classes in the scheduler map. Although these schedulers are configured on a single physical interface, they are independent from each other. Traffic on one logical interface unit does not affect the transmission priority, bandwidth allocation, or drop behavior on the other logical interface unit.

For a similar example, see "Example: Applying Scheduling and Shaping to VLANs" on page 312.

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.

```
set interfaces t3-1/0/0:1 per-unit-scheduler
set interfaces t3-1/0/0:1 encapsulation frame-relay
set interfaces t3-1/0/0:1 unit 0 dlci 100
set interfaces t3-1/0/0:1 unit 0 family inet address 10.1.1.0/24
set interfaces t3-1/0/0:1 unit 1 dlci 101
set interfaces t3-1/0/0:1 unit 1 family inet address 10.1.1.1/24
set class-of-service interfaces t3-1/0/0:1 unit 0 scheduler-map sched-map-logical-0
set class-of-service interfaces t3-1/0/0:1 unit 0 shaping-rate 10m
set class-of-service interfaces t3-1/0/0:1 unit 1 scheduler-map sched-map-logical-1
set class-of-service interfaces t3-1/0/0:1 unit 1 shaping-rate 20m
set class-of-service scheduler-maps sched-map-logical-0 forwarding-class best-effort
  scheduler sched-best-effort-0
set class-of-service scheduler-maps sched-map-logical-0 forwarding-class
  assured-forwarding scheduler sched-bronze-0
set class-of-service scheduler-maps sched-map-logical-0 forwarding-class
  expedited-forwarding scheduler sched-silver-0
set class-of-service scheduler-maps sched-map-logical-0 forwarding-class
  network-control scheduler sched-gold-0
set class-of-service scheduler-maps sched-map-logical-1 forwarding-class best-effort
  scheduler sched-best-effort-1
set class-of-service scheduler-maps sched-map-logical-1 forwarding-class
  assured-forwarding scheduler sched-bronze-1
set class-of-service scheduler-maps sched-map-logical-1 forwarding-class
  expedited-forwarding scheduler sched-silver-1
set class-of-service scheduler-maps sched-map-logical-1 forwarding-class
  network-control scheduler sched-gold-1
set class-of-service schedulers sched-best-effort-0 transmit-rate 4m
set class-of-service schedulers sched-bronze-0 transmit-rate 3m
set class-of-service schedulers sched-silver-0 transmit-rate 2m
set class-of-service schedulers sched-gold-0 transmit-rate 1m
set class-of-service schedulers sched-best-effort-1 transmit-rate 8m
set class-of-service schedulers sched-bronze-1 transmit-rate 6m
set class-of-service schedulers sched-silver-1 transmit-rate 4m
set class-of-service schedulers sched-gold-1 transmit-rate 2m
```

Copyright © 2019, Juniper Networks, Inc.

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see the *CLI User Guide*.

1. Configure the device interfaces.

```
[edit interfaces]
user@PE1# set t3-1/0/0:1 per-unit-scheduler
user@PE1# set t3-1/0/0:1 encapsulation frame-relay
user@PE1# set t3-1/0/0:1 unit 0 dlci 100
user@PE1# set t3-1/0/0:1 unit 0 family inet address 10.1.1.0/24
user@PE1# set t3-1/0/0:1 unit 1 dlci 101
user@PE1# set t3-1/0/0:1 unit 1 family inet address 10.1.1.1/24
```

2. Define the schedulers.

```
[edit class-of-service]
user@PE1# set schedulers sched-best-effort-0 transmit-rate 4m
user@PE1# set schedulers sched-bronze-0 transmit-rate 3m
user@PE1# set schedulers sched-silver-0 transmit-rate 2m
user@PE1# set schedulers sched-gold-0 transmit-rate 1m
user@PE1# set schedulers sched-best-effort-1 transmit-rate 8m
user@PE1# set schedulers sched-bronze-1 transmit-rate 6m
user@PE1# set schedulers sched-silver-1 transmit-rate 4m
user@PE1# set schedulers sched-gold-1 transmit-rate 2m
```

3. Define the scheduler maps.

```
[edit class-of-service]
user@PE1# set scheduler-maps sched-map-logical-0 forwarding-class best-effort
  scheduler sched-best-effort-0
user@PE1# set scheduler-maps sched-map-logical-0 forwarding-class
  assured-forwarding scheduler sched-bronze-0
user@PE1# set scheduler-maps sched-map-logical-0 forwarding-class
  expedited-forwarding scheduler sched-silver-0
user@PE1# set scheduler-maps sched-map-logical-0 forwarding-class
  network-control scheduler sched-gold-0
user@PE1# set scheduler-maps sched-map-logical-1 forwarding-class best-effort
  scheduler sched-best-effort-1
user@PE1# set scheduler-maps sched-map-logical-1 forwarding-class
  assured-forwarding scheduler sched-bronze-1
user@PE1# set scheduler-maps sched-map-logical-1 forwarding-class
  expedited-forwarding scheduler sched-silver-1
user@PE1# set scheduler-maps sched-map-logical-1 forwarding-class
  network-control scheduler sched-gold-1
```

4. Apply the scheduler maps and shaping rates to the logical interfaces.

```
[edit class-of-service]
user@PE1# set interfaces t3-1/0/0:1 unit 0 scheduler-map sched-map-logical-0
user@PE1# set interfaces t3-1/0/0:1 unit 0 shaping-rate 10m
user@PE1# set interfaces t3-1/0/0:1 unit 1 scheduler-map sched-map-logical-1
user@PE1# set interfaces t3-1/0/0:1 unit 1 shaping-rate 20m
```


Copyright © 2019, Juniper Networks, Inc.

Results

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

```
[edit interfaces]
user@PE1# show
t3-1/0/0:1 {
   encapsulation frame-relay;
   per-unit-scheduler;
}

[edit class-of-service]
user@PE1# show
interfaces {
   t3-1/0/0:1 {
      unit 0 {
         scheduler-map sched-map-logical-0;
         shaping-rate 10m;
      }
      unit 1 {
         scheduler-map sched-map-logical-1;
         shaping-rate 20m;
      }
   }
}
scheduler-maps {
   sched-map-logical-0 {
      forwarding-class best-effort scheduler sched-best-effort-0;
      forwarding-class assured-forwarding scheduler sched-bronze-0;
      forwarding-class expedited-forwarding scheduler sched-silver-0;
      forwarding-class network-control scheduler sched-gold-0;
   }
   sched-map-logical-1 {
      forwarding-class best-effort scheduler sched-best-effort-1;
      forwarding-class assured-forwarding scheduler sched-bronze-1;
      forwarding-class expedited-forwarding scheduler sched-silver-1;
      forwarding-class network-control scheduler sched-gold-1;
   }
}
schedulers {
   sched-best-effort-0 {
      transmit-rate 4m;
   }
   sched-bronze-0 {
      transmit-rate 3m;
   }
   sched-silver-0 {
      transmit-rate 2m;
   }
   sched-gold-0 {
      transmit-rate 1m;
```

Copyright © 2019, Juniper Networks, Inc.

```
    }
    sched-best-effort-1 {
      transmit-rate 8m;
    }
    sched-bronze-1 {
      transmit-rate 6m;
    }
    sched-silver-1 {
      transmit-rate 4m;
    }
    sched-gold-1 {
      transmit-rate 2m;
    }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Related Documentation

- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300
- Example: Applying Scheduling and Shaping to VLANs on page 312
- **per-unit-scheduler on page 1226**

Example: Applying Scheduling and Shaping to VLANs

This example shows how to apply schedulers to individual logical interfaces.

- Requirements on page 312
- Overview on page 312
- Configuration on page 313

Requirements

This example uses the following hardware and software components:

- Junos OS Release 7.4 or later running on router line cards that support Intelligent Queuing (IQ).
- Junos OS Release 13.2 or later running on MX Series routers containing 16x10GE MPC or MPC3E line cards.
- Junos OS Release 13.3 or later running on MX Series routers containing MPC4E line cards.
- Junos OS Release 15.1 or later running on MX Series routers containing MPC6E line cards.

Overview

By default, output scheduling is not enabled on logical interfaces. Logical interfaces without shaping configured share a default scheduler. *Logical interface scheduling* (also

Copyright © 2019, Juniper Networks, Inc.

called *per-unit scheduling*) allows you to enable multiple output queues on a logical interface and associate customized scheduling and shaping for each queue.

To enable per-unit scheduling, include the **per-unit-scheduler** statement at the **[edit interfaces *interface name*]** hierarchy level. When per-unit schedulers are enabled, you can define dedicated schedulers for logical interfaces by including the **scheduler-map** statement at the **[edit class-of-service interfaces *interface name* unit *logical unit number*]** hierarchy level. Alternatively, you can include the **scheduler-map** statement at the **[edit class-of-service traffic-control-profiles *traffic control profile name*]** hierarchy level and then include the **output-traffic-control-profile** statement at the **[edit class-of-service interfaces *interface name* unit *logical unit number*]** hierarchy level.

This example shows how to define schedulers for logical interfaces through the use of traffic control profiles.

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.

```
set interfaces xe-9/0/3 per-unit-scheduler
set interfaces xe-9/0/3 vlan-tagging
set interfaces xe-9/0/3 unit 1 vlan-id 101
set interfaces xe-9/0/3 unit 1 family inet address 10.1.1.1/24
set interfaces xe-9/0/3 unit 2 vlan-id 102
set interfaces xe-9/0/3 unit 2 family inet address 10.2.1.1/24
set class-of-service classifiers inet-precedence c8 forwarding-class be loss-priority low
  code-points 000
set class-of-service classifiers inet-precedence c8 forwarding-class ef loss-priority low
  code-points 001
set class-of-service classifiers inet-precedence c8 forwarding-class af loss-priority low
  code-points 010
set class-of-service classifiers inet-precedence c8 forwarding-class nc loss-priority low
  code-points 011
set class-of-service classifiers inet-precedence c8 forwarding-class be1 loss-priority low
  code-points 100
set class-of-service classifiers inet-precedence c8 forwarding-class ef1 loss-priority low
  code-points 101
set class-of-service classifiers inet-precedence c8 forwarding-class af1 loss-priority low
  code-points 110
set class-of-service classifiers inet-precedence c8 forwarding-class nc1 loss-priority low
  code-points 111
set class-of-service forwarding-classes queue 0 be
set class-of-service forwarding-classes queue 1 ef
set class-of-service forwarding-classes queue 2 af
set class-of-service forwarding-classes queue 3 nc
set class-of-service forwarding-classes queue 4 be1
set class-of-service forwarding-classes queue 5 ef1
set class-of-service forwarding-classes queue 6 af1
set class-of-service forwarding-classes queue 7 nc1
set class-of-service traffic-control-profiles tcp_ifd shaping-rate 2500000000
```

Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service traffic-control-profiles tcp_ifd overhead-accounting bytes -20
set class-of-service traffic-control-profiles tcp_gold scheduler-map gold
set class-of-service traffic-control-profiles tcp_gold shaping-rate 2500000000
set class-of-service traffic-control-profiles tcp_gold overhead-accounting bytes -20
set class-of-service traffic-control-profiles tcp_gold guaranteed-rate 1g
set class-of-service traffic-control-profiles tcp_silver scheduler-map silver
set class-of-service traffic-control-profiles tcp_silver shaping-rate 1g
set class-of-service traffic-control-profiles tcp_silver overhead-accounting bytes -20
set class-of-service traffic-control-profiles tcp_silver guaranteed-rate 500m
set class-of-service interfaces xe-9/0/3 output-traffic-control-profile tcp_ifd
set class-of-service interfaces xe-9/0/3 unit 1 output-traffic-control-profile tcp_gold
set class-of-service interfaces xe-9/0/3 unit 2 output-traffic-control-profile tcp_silver
set class-of-service scheduler-maps gold forwarding-class be1 scheduler gold_internet
set class-of-service scheduler-maps gold forwarding-class ef1 scheduler gold_video
set class-of-service scheduler-maps gold forwarding-class af1 scheduler gold_voice
set class-of-service scheduler-maps gold forwarding-class nc1 scheduler gold_reserved
set class-of-service scheduler-maps silver forwarding-class be scheduler silver_internet
set class-of-service scheduler-maps silver forwarding-class ef scheduler silver_video
set class-of-service scheduler-maps silver forwarding-class af scheduler silver_voice
set class-of-service scheduler-maps silver forwarding-class nc scheduler silver_reserved
set class-of-service schedulers gold_internet excess-rate percent 40
set class-of-service schedulers gold_internet buffer-size percent 20
set class-of-service schedulers gold_internet priority low
set class-of-service schedulers gold_video transmit-rate percent 50
set class-of-service schedulers gold_video buffer-size percent 50
set class-of-service schedulers gold_voice shaping-rate percent 10
set class-of-service schedulers gold_voice buffer-size percent 10
set class-of-service schedulers gold_voice priority strict-high
set class-of-service schedulers gold_reserved excess-rate percent 20
set class-of-service schedulers gold_reserved buffer-size percent 10
set class-of-service schedulers gold_reserved priority low
set class-of-service schedulers silver_internet excess-rate percent 40
set class-of-service schedulers silver_internet buffer-size percent 20
set class-of-service schedulers silver_internet priority low
set class-of-service schedulers silver_video transmit-rate percent 50
set class-of-service schedulers silver_video buffer-size percent 50
set class-of-service schedulers silver_voice shaping-rate percent 10
set class-of-service schedulers silver_voice buffer-size percent 10
set class-of-service schedulers silver_voice priority strict-high
set class-of-service schedulers silver_reserved excess-rate percent 20
set class-of-service schedulers silver_reserved buffer-size percent 10
set class-of-service schedulers silver_reserved priority low
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see the *CLI User Guide*.

1. Configure the device interfaces.

```
[edit interfaces]
user@PE1# set xe-9/0/3 per-unit-scheduler
user@PE1# set xe-9/0/3 vlan-tagging
user@PE1# set xe-9/0/3 unit 1 vlan-id 101
user@PE1# set xe-9/0/3 unit 1 family inet address 10.1.1.1/24
```


Copyright © 2019, Juniper Networks, Inc.

```
user@PE1# set xe-9/0/3 unit 2 vlan-id 102
user@PE1# set xe-9/0/3 unit 2 family inet address 10.2.1.1/24
```

2. Configure the classifiers.

```
[edit class-of-service]
user@PE1# set classifiers inet-precedence c8 forwarding-class be loss-priority low
  code-points 000
user@PE1# set classifiers inet-precedence c8 forwarding-class ef loss-priority low
  code-points 001
user@PE1# set classifiers inet-precedence c8 forwarding-class af loss-priority low
  code-points 010
user@PE1# set classifiers inet-precedence c8 forwarding-class nc loss-priority low
  code-points 011
user@PE1# set classifiers inet-precedence c8 forwarding-class be1 loss-priority low
  code-points 100
user@PE1# set classifiers inet-precedence c8 forwarding-class ef1 loss-priority low
  code-points 101
user@PE1# set classifiers inet-precedence c8 forwarding-class af1 loss-priority low
  code-points 110
user@PE1# set classifiers inet-precedence c8 forwarding-class nc1 loss-priority low
  code-points 111
```

3. Configure the forwarding classes.

```
[edit class-of-service]
user@PE1# set forwarding-classes queue 0 be
user@PE1# set forwarding-classes queue 1 ef
user@PE1# set forwarding-classes queue 2 af
user@PE1# set forwarding-classes queue 3 nc
user@PE1# set forwarding-classes queue 4 be1
user@PE1# set forwarding-classes queue 5 ef1
user@PE1# set forwarding-classes queue 6 af1
user@PE1# set forwarding-classes queue 7 nc1
```

4. Configure the traffic control profiles.

```
[edit class-of-service]
user@PE1# set traffic-control-profiles tcp_ifd shaping-rate 2500000000
user@PE1# set traffic-control-profiles tcp_ifd overhead-accounting bytes -20
user@PE1# set traffic-control-profiles tcp_gold scheduler-map gold
user@PE1# set traffic-control-profiles tcp_gold shaping-rate 2500000000
user@PE1# set traffic-control-profiles tcp_gold overhead-accounting bytes -20
user@PE1# set traffic-control-profiles tcp_gold guaranteed-rate 1g
user@PE1# set traffic-control-profiles tcp_silver scheduler-map silver
user@PE1# set traffic-control-profiles tcp_silver shaping-rate 1g
user@PE1# set traffic-control-profiles tcp_silver overhead-accounting bytes -20
user@PE1# set traffic-control-profiles tcp_silver guaranteed-rate 500m
```

5. Map the traffic control profiles to their respective physical or logical interface.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service]
user@PE1# set interfaces xe-9/0/3 output-traffic-control-profile tcp_ifd
user@PE1# set interfaces xe-9/0/3 unit 1 output-traffic-control-profile tcp_gold
user@PE1# set interfaces xe-9/0/3 unit 2 output-traffic-control-profile tcp_silver
```

6. Configure the scheduler maps.

```
[edit class-of-service]
user@PE1# set scheduler-maps gold forwarding-class be1 scheduler gold_internet
user@PE1# set scheduler-maps gold forwarding-class ef1 scheduler gold_video
user@PE1# set scheduler-maps gold forwarding-class af1 scheduler gold_voice
user@PE1# set scheduler-maps gold forwarding-class nc1 scheduler gold_reserved
user@PE1# set scheduler-maps silver forwarding-class be scheduler silver_internet
user@PE1# set scheduler-maps silver forwarding-class ef scheduler silver_video
user@PE1# set scheduler-maps silver forwarding-class af scheduler silver_voice
user@PE1# set scheduler-maps silver forwarding-class nc scheduler silver_reserved
```

7. Configure the schedulers.

```
[edit class-of-service]
user@PE1# set schedulers gold_internet excess-rate percent 40
user@PE1# set schedulers gold_internet buffer-size percent 20
user@PE1# set schedulers gold_internet priority low
user@PE1# set schedulers gold_video transmit-rate percent 50
user@PE1# set schedulers gold_video buffer-size percent 50
user@PE1# set schedulers gold_voice shaping-rate percent 10
user@PE1# set schedulers gold_voice buffer-size percent 10
user@PE1# set schedulers gold_voice priority strict-high
user@PE1# set schedulers gold_reserved excess-rate percent 20
user@PE1# set schedulers gold_reserved buffer-size percent 10
user@PE1# set schedulers gold_reserved priority low
user@PE1# set schedulers silver_internet excess-rate percent 40
user@PE1# set schedulers silver_internet buffer-size percent 20
user@PE1# set schedulers silver_internet priority low
user@PE1# set schedulers silver_video transmit-rate percent 50
user@PE1# set schedulers silver_video buffer-size percent 50
user@PE1# set schedulers silver_voice shaping-rate percent 10
user@PE1# set schedulers silver_voice buffer-size percent 10
user@PE1# set schedulers silver_voice priority strict-high
user@PE1# set schedulers silver_reserved excess-rate percent 20
user@PE1# set schedulers silver_reserved buffer-size percent 10
user@PE1# set schedulers silver_reserved priority low
```

Results

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

```
user@PE1# show interfaces
```


Copyright © 2019, Juniper Networks, Inc.

```
interfaces {
    xe-9/0/3 {
        per-unit-scheduler;
        vlan-tagging;
        unit 1 {
            vlan-id 101;
            family inet {
                address 10.1.1.1/24;
            }
        }
        unit 2 {
            vlan-id 102;
            family inet {
                address 10.2.1.1/24;
            }
        }
    }
}
```

user@PE1# show class-of-service

```
class-of-service {
    classifiers {
        inet-precedence c8 {
            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 be1 {
                loss-priority low code-points 100;
            }
            forwarding-class ef1 {
                loss-priority low code-points 101;
            }
            forwarding-class af1 {
                loss-priority low code-points 110;
            }
            forwarding-class nc1 {
                loss-priority low code-points 111;
            }
        }
    }
    forwarding-classes {
        queue 0 be;
        queue 1 ef;
        queue 2 af;
        queue 3 nc;
        queue 4 be1;
        queue 5 ef1;
        queue 6 af1;
        queue 7 nc1;
    }
    traffic-control-profiles {
```

Copyright © 2019, Juniper Networks, Inc.

```
        tcp_ifd {
            shaping-rate 2500000000;
            overhead-accounting bytes -20;
        }
        tcp_gold {
            scheduler-map gold;
            shaping-rate 2500000000;
            overhead-accounting bytes -20;
            guaranteed-rate 1g;
        }
        tcp_silver {
            scheduler-map silver;
            shaping-rate 1g;
            overhead-accounting bytes -20;
            guaranteed-rate 500m;
        }
    }
    interfaces {
        xe-9/0/3 {
            output-traffic-control-profile tcp_ifd;
            unit 1 {
                output-traffic-control-profile tcp_gold;
            }
            unit 2 {
                output-traffic-control-profile tcp_silver;
            }
        }
    }
    scheduler-maps {
        gold {
            forwarding-class be1 scheduler gold_internet;
            forwarding-class ef1 scheduler gold_video;
            forwarding-class af1 scheduler gold_voice;
            forwarding-class nc1 scheduler gold_reserved;
        }
        silver {
            forwarding-class be scheduler silver_internet;
            forwarding-class ef scheduler silver_video;
            forwarding-class af scheduler silver_voice;
            forwarding-class nc scheduler silver_reserved;
        }
    }
    schedulers {
        gold_internet {
            excess-rate percent 40;
            buffer-size percent 20;
            priority low;
        }
        gold_video {
            transmit-rate percent 50;
            buffer-size percent 50;
        }
        gold_voice {
            shaping-rate percent 10;
            buffer-size percent 10;
            priority strict-high;
        }
        gold_reserved {
            excess-rate percent 20;
            buffer-size percent 10;
```


Copyright © 2019, Juniper Networks, Inc.

```
            priority low;
        }
        silver_internet {
            excess-rate percent 40;
            buffer-size percent 20;
            priority low;
        }
        silver_video {
            transmit-rate percent 50;
            buffer-size percent 50;
        }
        silver_voice {
            shaping-rate percent 10;
            buffer-size percent 10;
            priority strict-high;
        }
        silver_reserved {
            excess-rate percent 20;
            buffer-size percent 10;
            priority low;
        }
    }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Related Documentation

- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300
- Example: Applying Scheduler Maps and Shaping Rate to DLCIs on page 308
- **per-unit-scheduler on page 1226**

Example: Limiting Egress Traffic on an Interface Using Port Shaping for CoS

This example shows how using port shaping as a form of class of service (CoS) enables you to limit traffic on an interface, so that you can control the amount of traffic passing through the interface.

- Requirements on page 319
- Overview on page 320
- Configuration on page 320
- Verification on page 325

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Copyright © 2019, Juniper Networks, Inc.

Overview

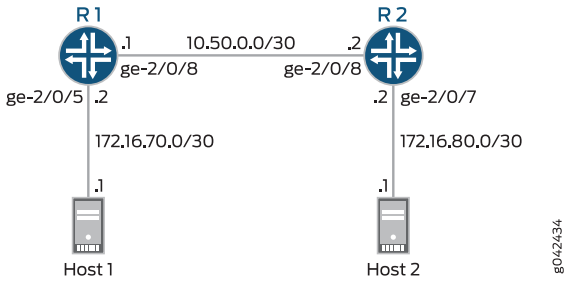
The purpose of this example is to demonstrate how port shaping enables you to shape the traffic passing through an interface to a rate that is less than the line rate for that interface. When you configure port shaping on an interface, you are essentially specifying a value that indicates the maximum amount of traffic that can pass through the interface. This value must be less than the maximum bandwidth for that interface. When you configure port shaping, you can specify either the maximum rate at which traffic can pass through the interface or as a percentage of the bandwidth of the interface.

In this example the port shaping is done on Device R1. The information required for implementing port shaping on Device R2 is not included in this example. However, you can use the port shaping information in Device R1 (making changes for the interfaces used) and apply it to Device R2 to achieve port shaping on Device R2.

Topology

This example uses the topology in Figure 34 on page 320.

Figure 34: Port Shaping Scenario



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.

Device R1 Using Only Class of Service

```
set interfaces ge-2/0/5 description to-Host
set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
set interfaces ge-2/0/8 description to-R2
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.13.1/32
set class-of-service interfaces ge-2/0/8 shaping-rate 160k
set protocols ospf area 0.0.0.0 interface ge-2/0/5.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```


Copyright © 2019, Juniper Networks, Inc.

Device R1 Using Traffic Control Profiles and Class of Service

```
set interfaces ge-2/0/5 description to-Host
set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
set interfaces ge-2/0/8 description to-R2
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.13.1/32
set class-of-service traffic-control-profiles output shaping-rate 160k
set class-of-service traffic-control-profiles output shaping-rate burst-size 30k
set class-of-service interfaces ge-2/0/8 output-traffic-control-profile output
set protocols ospf area 0.0.0.0 interface ge-2/0/5.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Device R2

```
set interfaces ge-2/0/7 description to-Host
set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
set interfaces ge-2/0/8 description to-R1
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.14.1/32
set protocols ospf area 0.0.0.0 interface ge-2/0/7.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

You can configure port shaping on network interfaces, aggregated Ethernet interfaces (also known as link aggregation groups (LAGs)), and loopback interfaces.

To configure Device R1:

1. Configure the device interfaces.

```
[edit]
user@R1# set interfaces ge-2/0/5 description to-Host
user@R1# set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
user@R1# set interfaces ge-2/0/8 description to-R2
user@R1# set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
user@R1# set interfaces lo0 unit 0 description looback-interface
user@R1# set interfaces lo0 unit 0 family inet address 192.168.13.1/32
```

2. Configure port shaping using only class of service.

```
[edit]
user@R1# set class-of-service interfaces ge-2/0/8 shaping-rate 160k
```

3. Configure port shaping using traffic control profiles and class of service.

Copyright © 2019, Juniper Networks, Inc.

NOTE: If you configure a fixed shaping rate, you can configure an optional burst size in bytes. If you configure the shaping rate as a percentage, the **burst-size** option is not allowed.

```
[edit]
user@R1# set class-of-service traffic-control-profiles output shaping-rate 160k
user@R1# set class-of-service traffic-control-profiles output shaping-rate burst-size
  30k
user@R1# set class-of-service interfaces ge-2/0/8 output-traffic-control-profile
  output
```

4. Configure OSPF.

```
[edit]
user@R1# set protocols ospf area 0.0.0.0 interface ge-2/0/5.0 passive
user@R1# set protocols ospf area 0.0.0.0 interface lo0.0 passive
user@R1# set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

To configure Device R2:

1. Configure the device interfaces.

```
[edit interfaces]
set interfaces ge-2/0/7 description to-Host
set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
set interfaces ge-2/0/8 description to-R1
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.14.1/32
```

2. Configure OSPF.

```
[edit ]
user@R1# set protocols ospf area 0.0.0.0 interface ge-2/0/7.0 passive
user@R1# set protocols ospf area 0.0.0.0 interface lo0.0 passive
user@R1# set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Results

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

Results for R1

Copyright © 2019, Juniper Networks, Inc.

```
user@R1# show interfaces
ge-2/0/5 {
   description to-Host;
   unit 0 {
      family inet {
         address 172.16.70.2/30;
      }
   }
}
ge-2/0/8 {
   description to-R2;
   unit 0 {
      family inet {
         address 10.50.0.1/30;
      }
   }
}
lo0 {
   unit 0 {
      description looback-interface;
      family inet {
         address 192.168.13.1/32;
      }
   }
}
```

Configuring Port Shaping Using only Class-of-Service

```
user@R1# show class-of-service
interfaces {
   ge-2/0/8 {
      shaping-rate 160k;
   }
}
```

Configuring Port Shaping Using Traffic Control Profiles and Class of Service

```
user@R1# show class-of-service
traffic-control-profiles {
   output {
      shaping-rate 160k burst-size 30k;
   }
}
interfaces {
   ge-2/0/8 {
      output-traffic-control-profile output;
   }
}
```

```
user@R1# show protocols ospf
area 0.0.0.0 {
   interface ge-2/0/5.0 {
      passive;
   }
   interface lo0.0 {
```

Copyright © 2019, Juniper Networks, Inc.

```
    passive;
  }
  interface ge-2/0/8.0;
}
```

If you are done configuring Device R1, enter **commit** from configuration mode.

Results for R2

```
user@R2# show interfaces
ge-2/0/7 {
  description to-Host;
  unit 0 {
    family inet {
      address 172.16.80.2/30;
    }
  }
}
ge-2/0/8 {
  description to-R1;
  unit 0 {
    family inet {
      address 10.50.0.2/30;
    }
  }
}
lo0 {
  unit 0 {
    description looback-interface;
    family inet {
      address 192.168.14.1/32;
    }
  }
}
```

```
user@R2# show protocols ospf
area 0.0.0.0 {
  interface ge-2/0/7.0 {
    passive;
  }
  interface lo0.0 {
    passive;
  }
  interface ge-2/0/8.0;
}
```

If you are done configuring Device R2, enter **commit** from configuration mode.

Copyright © 2019, Juniper Networks, Inc.

Verification

Confirm that the configuration is working properly.

- Clearing the Counters on page 325
- Sending TCP Traffic into the Network and Monitoring the Port Shaping on page 325

Clearing the Counters

Purpose Confirm that the interface counters are cleared.

Action On Device R1, run the **clear interfaces statistics ge-2/0/8** command to reset the interface statistics to 0.

```
user@R1> clear interfaces statistics ge-2/0/8
```

Sending TCP Traffic into the Network and Monitoring the Port Shaping

Purpose Make sure that the traffic is rate-limited on the output interface (ge-2/0/8) on Device R1 by sending traffic into the network using the host connected to Device R1.

Action

1. Use a traffic generator to send several continuous streams of TCP packets with a source port of 80.

   The -s flag sets the source port. The -k flag causes the source port to remain steady at 80 instead of incrementing. The -d flag sets the packet size. The -c flag sets the packet count to be sent.

   The destination IP address of 172.16.80.1 represents a user that is downstream of Device R2. The user has requested a webpage from the host (the webserver emulated by the traffic generator), and the packets are sent in response to the request.

   NOTE: Remember in this example the port shaping has been set to 160 Kbps.

   ```
   [user@host]# hping 172.16.80.1 -s 80 -k -d 1500 -c 20 &
   hping 172.16.80.1 -s 80 -k -d 1500 -c 20 &
   .
   .
   .
   ```

2. On Device R1, check the interface counters by using the **show interfaces extensive ge-2/0/8** command.

   ```
   user@R1> show interfaces extensive ge-2/0/8
   ```

Copyright © 2019, Juniper Networks, Inc.

```
Queue counters:      Queued packets  Transmitted packets     Dropped packets

  0                          17244                 3741           13470
  1                             13                   13               0
  2                              0                    0               0
  3                         149363               149363               0
Queue number:         Mapped forwarding classes
  0                   best-effort
  1                   expedited-forwarding
  2                   assured-forwarding
  3                   network-control
```

Meaning In the output you can see that 13470 packets have been dropped. These are the packets that exceeded the 160 Kbps shaping rate configured on ge-2/0/8.

Related Documentation

- *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide*
- *Example: Configuring a Two-Rate Three-Color Policer*

Configuring Input Shaping Rates for Both Physical and Logical Interfaces

You can apply input shaping rates to both the physical interface and its logical interfaces. The rate specified at the physical level is distributed among the logical interfaces based on their input shaping-rate ratio.

To configure an input shaper on the physical interface:

- Specify the phyiscal interface and associated shaping rate.

```
[edit]
user@host# edit class-of-service interfaces interface-name
user@host# set input-shaping-rate rate
```

To configure an input shaper on the logical interface:

- Specify the physical and logical interface names and associated shaping rate.

```
[edit]
user@host# edit class-of-service interfaces interface-name unit logical-unit-number
user@host# set input-shaping-rate (percent percentage | rate)
```

For each logical interface, you can specify a percentage of the physical rate or an actual rate. The Junos OS software converts actual rates into percentages of the physical rate.

Related Documentation

- Configuring Shaping on 10-Gigabit Ethernet IQ2 PICs on page 824

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 8

Setting Transmission Order with Scheduler Priorities and Hierarchical Scheduling

- Priority Scheduling Overview on page 327
- Platform Support for Priority Scheduling on page 329
- Configuring Schedulers for Priority Scheduling on page 330
- Associating Schedulers with Fabric Priorities on page 331
- Hierarchical Class of Service Overview on page 333
- Hierarchical Class of Service Network Scenarios on page 336
- Understanding Hierarchical Scheduling on page 337
- Priority Propagation in Hierarchical Scheduling on page 340
- Configuring Hierarchical Schedulers for CoS on page 342
- Hierarchical Schedulers and Traffic Control Profiles on page 343
- Example: Building a Four-Level Hierarchy of Schedulers on page 345

Priority Scheduling Overview

The Junos OS supports multiple levels of transmission priority, which in order of increasing priority are **low**, **medium-low**, **medium-high**, and **high**, and **strict-high**. This allows the software to service higher-priority queues before lower-priority queues.

Priority scheduling determines the order in which an output interface transmits traffic from its queues, thus ensuring that queues containing important traffic are provided better access to the outgoing interface. Junos OS accomplishes priority scheduling by examining the assigned priority of each individual queue and whether each individual queue is within its defined bandwidth profile. Junos OS determines whether an individual queue is within its bandwidth profile by comparing, at regular intervals, the amount of data transmitted by the queue against the amount of bandwidth allocated to it by the configured scheduler transmission rate (transmit-rate) defined at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level. When the transmitted amount is less than the allocated amount, the queue is considered to be *in profile*. A queue is *out of profile* when its transmitted amount is larger than its allocated amount.

Copyright © 2019, Juniper Networks, Inc.

The queues for a given output physical interface (or output logical interface if per-unit scheduling is enabled on that interface) are divided into sets based on their priority. Any such set contains queues of the same priority.

Junos OS traverses the sets in descending order of priority. If at least one of the queues in the set has a packet to transmit, the software selects that set. A queue from the set is selected based on the weighted round robin (WRR) algorithm, which operates within the set.

The Junos OS performs priority queuing using the following steps:

1. The software locates all high-priority queues that are currently in profile. These queues are serviced first in a weighted round-robin fashion.
2. The software locates all medium-high priority queues that are currently in profile. These queues are serviced second in a weighted round-robin fashion.
3. The software locates all medium-low priority queues that are currently in profile. These queues are serviced third in a weighted round-robin fashion.
4. The software locates all low-priority queues that are currently in profile. These queues are serviced fourth in a weighted round-robin fashion.
5. The software locates all high-priority queues that are currently out of profile and are not rate limited. The weighted round-robin algorithm is applied to these queues for servicing.
6. The software locates all medium-high priority queues that are currently out of profile and are not rate limited. The weighted round-robin algorithm is applied to these queues for servicing.
7. The software locates all medium-low priority queues that are currently out of profile and are not rate limited. The weighted round-robin algorithm is applied to these queues for servicing.
8. The software locates all low-priority queues that are currently out of profile and are also not rate limited. These queues are serviced last in a weighted round-robin manner.

Strict-High Priority Configuration Overview

You can configure one queue per interface to have **strict-high** priority, which works the same as **high** priority, but provides unlimited transmission bandwidth. As long as the queue with **strict-high** priority has traffic to send, it receives precedence over all other queues, except queues with **high** priority. Queues with **strict-high** and **high** priority take turns transmitting packets until the **strict-high** queue is empty, the **high** priority queues are empty, or the **high** priority queues run out of bandwidth credit. Only when these conditions are met can lower priority queues send traffic.

When you configure a queue to have **strict-high** priority, you do not need to include the **transmit-rate** statement in the queue configuration at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level because the transmission rate of a **strict-high** priority queue is not limited by the WRR configuration. If you do configure a transmission rate on a **strict-high** priority queue, it does not affect the WRR operation. The transmission rate

Copyright © 2019, Juniper Networks, Inc.

does, however, affect the calculation of the delay buffer and also serves as a placeholder in the output of commands such as the **show interface queue** command.

strict-high priority queues might starve **low** priority queues, and under certain circumstances might limit **high** priority queues. The **high** priority allows you to protect traffic classes from being starved by traffic in a **strict-high** queue. For example, a network-control queue might require a small bandwidth allocation (say, 5 percent). You can assign **high** priority to this queue to prevent it from being underserved.

A queue with **strict-high** priority supersedes bandwidth guarantees for queues with lower priority; therefore, we recommend that you use the **strict-high** priority to ensure proper ordering of special traffic, such as voice traffic. You can preserve bandwidth guarantees for queues with lower priority by allocating to the queue with **strict-high** priority only the amount of bandwidth that it generally requires by applying the **rate-limit**option to the **strict-high** queue's transmission rate. For example, consider the following allocation of transmission bandwidth:

- Q0 BE—20 percent, low priority
- Q1 EF—30 percent, strict-high priority
- Q2 AF—40 percent, low priority
- Q3 NC—10 percent, low priority

This bandwidth allocation assumes that, in general, the EF forwarding class requires only 30 percent of an interface's transmission bandwidth. However, if short bursts of traffic are received on the EF forwarding class, and the **rate-limit**option is not applied, 100 percent of the bandwidth is given to the EF forwarding class because of the **strict-high** setting.

Related Documentation

- How Schedulers Define Output Queue Properties on page 254

Platform Support for Priority Scheduling

Hardware platforms support queue priorities in different ways:

- On all platforms, you can configure one queue per interface to have strict-high priority.
- Strict-high priority works differently on Multiservices and Services PIC link services IQ (**lsq-**) interfaces. For link services IQ interfaces, a queue with strict-high priority might starve all the other queues. For more information, see the *Junos OS Services Interfaces Library for Routing Devices*.
- The priority levels you configure map to hardware priority levels. These priority mappings depend on the FPC type in which the PIC is installed.

Table 41 on page 330 shows the priority mappings by FPC type. Note, for example, that on Juniper Networks M320 Multiservice Edge Routers FPCs, T Series Core Routers FPCs and T Series Enhanced FPCs, the software priorities **medium-low** and **medium-high** behave similarly because they map to the same hardware priority level.

Copyright © 2019, Juniper Networks, Inc.

Table 41: Scheduling Priority Mappings by FPC Type

Priority Levels	Mappings for FPCs	Mappings for M320 FPCs and T Series Enhanced FPCs	Mappings for M120 FEBs
low	0	0	0
medium-low	0	1	1
medium-high	1	1	2
high	1	2	3
strict-high (full interface bandwidth)	1	2	3

Related Documentation

- How Schedulers Define Output Queue Properties on page 254
- Configuring Schedulers for Priority Scheduling on page 330

Configuring Schedulers for Priority Scheduling

This topic describes how to configure priority scheduling.

```
[edit class-of-service schedulers scheduler-name]
priority priority-level;
```

The priority level can be **low**, **medium-low**, **medium-high**, **high**, or **strict-high**. The priorities map to numeric priorities in the underlying hardware. In some cases, different priorities behave similarly, because two software priorities behave differently only if they map to two distinct hardware priorities. For more information, see "Platform Support for Priority Scheduling" on page 329.

Higher-priority queues transmit packets ahead of lower priority queues as long as the higher-priority forwarding classes retain enough bandwidth credit. When you configure a higher-priority queue with a significant fraction of the transmission bandwidth, the queue might lock out (or *starve*) lower priority traffic.

In the following example procedure, you create a scheduler, configure the mapping between the scheduler and the forwarding class, and assign the scheduler to an interface.

1. Configure a scheduler, **be-sched**, with **medium-low** priority.

   ```
   [edit]
   user@host# edit class-of-service schedulers be-sched
   user@host# set priority medium-low
   ```

2. Configure a scheduler map, **be-map**, that associates **be-sched** with the **best-effort** forwarding class.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service]
user@host# set scheduler-maps be-map forwarding-class best-effort scheduler
  be-sched
```

3. Assign the **be-map** scheduler map to a Gigabit Ethernet interface, **ge-0/0/0**.

```
[edit class-of-service]
user@host# set interfaces ge-0/0/0 scheduler-map be-map
```

4. Verify your configuration.

```
[edit class-of-service]
user@host# show
```

```
schedulers {
    be-sched {
        priority medium-low;
    }
}
scheduler-maps {
    be-map {
        forwarding-class best-effort scheduler be-sched;
    }
}
ge-0/0/0 {
    scheduler-map be-map;
}
```

5. Save your configuration.

```
[edit class-of-service]
user@host# commit
```

Related Documentation

- Priority Scheduling Overview on page 327
- How Schedulers Define Output Queue Properties on page 254
- Platform Support for Priority Scheduling on page 329

Associating Schedulers with Fabric Priorities

On Juniper Networks M320 routers, MX Series routers, T Series routers and EX Series switches only, you can associate a scheduler with a class of traffic that has a specific priority while transiting the fabric. Traffic transiting the fabric can have two priority values: **low** or **high**. To associate a scheduler with a fabric priority, include the **priority** and **scheduler** statements at the **[edit class-of-service fabric scheduler-map]** hierarchy level:

```
[edit class-of-service fabric scheduler-map]
priority (high | low) scheduler scheduler-name;
```

Copyright © 2019, Juniper Networks, Inc.

NOTE: For a scheduler that you associate with a fabric priority, include only the **drop-profile-map** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level. You cannot include the **buffer-size**, **transmit-rate**, and **priority** statements at that hierarchy level.

Example: Associating a Scheduler with a Fabric Priority

Associate a scheduler with a class of traffic that has a specific priority while transiting the fabric:

```
[edit class-of-service]
schedulers {
   fab-be-scheduler {
      drop-profile-map loss-priority low protocol any drop-profile fab-profile-1;
      drop-profile-map loss-priority high protocol any drop-profile fab-profile-2;
   }
   fab-ef-scheduler {
      drop-profile-map loss-priority low protocol any drop-profile fab-profile-3;
      drop-profile-map loss-priority high protocol any drop-profile fab-profile-4;
   }
}
drop-profiles {
   fab-profile-1 {
      fill-level 100 drop-probability 100;
      fill-level 85 drop-probability 50;
   }
   fab-profile-2 {
      fill-level 100 drop-probability 100;
      fill-level 95 drop-probability 50;
   }
   fab-profile-3 {
      fill-level 75 drop-probability 100;
      fill-level 95 drop-probability 50;
   }
   fab-profile-4 {
      fill-level 100 drop-probability 100;
      fill-level 80 drop-probability 50;
   }
}
fabric {
   scheduler-map {
      priority low scheduler fab-be-scheduler;
      priority high scheduler fab-ef-scheduler;
   }
}
```

Related Documentation

- Forwarding Classes and Fabric Priority Queues on page 234

Copyright © 2019, Juniper Networks, Inc.

Hierarchical Class of Service Overview

Hierarchical class of service (HCoS) is the ability to apply traffic schedulers and shapers to a hierarchy of *scheduler nodes*. Each level of the scheduler hierarchy can be used to shape traffic based on different criteria such as application, user, VLAN, and physical port.

This allows you to support the requirements of different services, applications, and users on the same physical device and physical infrastructure.

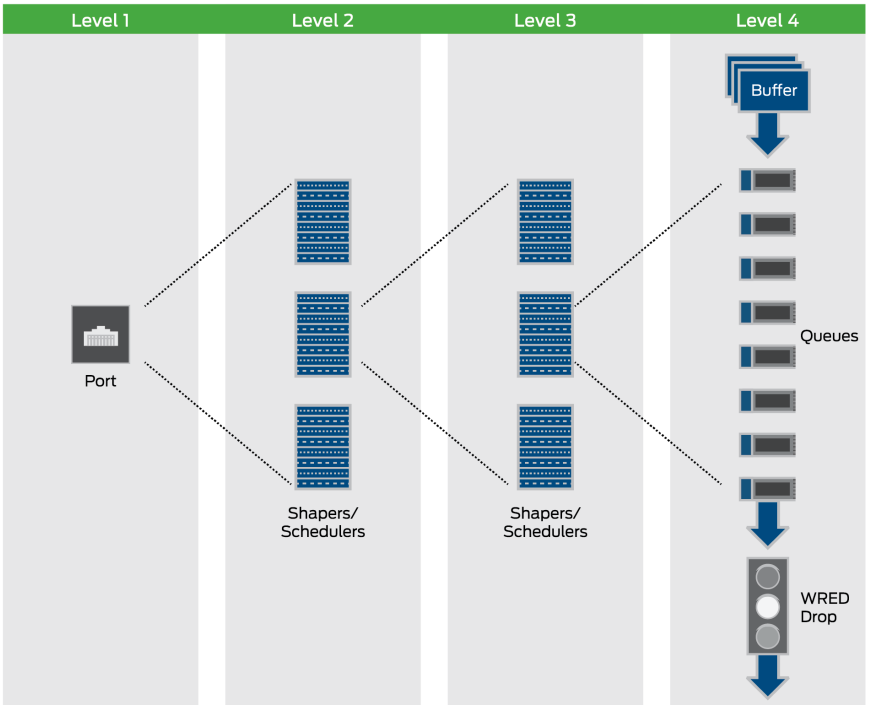
HCoS is implemented primarily using traffic classifiers at the ingress and hierarchical schedulers and shapers at the egress.

A classifier is a filter that labels traffic at the device ingress based on configurable parameters such as application or destination. Traffic is classified into what is called a forwarding equivalence class (FEC). The FEC defines a class of traffic that receives common treatment.

Schedulers, and their associated shapers, is the function that controls the traffic bandwidth, jitter (delay variation), and packet loss priority at the egress of the device.

Hierarchical schedulers are used to apply multiple levels of scheduling and shaping with each level applied to different classifications such as forwarding equivalence class, VLAN, and physical interface (port) as shown in Figure 35 on page 333.

Figure 35: Hierarchical Scheduling Architecture



Copyright © 2019, Juniper Networks, Inc.

NOTE: Hierarchical class of service is also referred to as Hierarchical Quality of Service (HQoS) in other vendor's documentation.

A typical application of HCoS is to configure multiple levels of egress schedulers and shapers, at the subscriber edge, using dynamic profiles to provide traffic shaping and prioritization at the subscriber VLAN level and for multiple classes of traffic.

Dynamic profiles are a mechanism that allows you to dynamically apply schedulers and shapers to individual subscribers or groups of subscribers.

To learn more about HCoS, the following topics are very helpful:

- Junos CoS on MX Series 5G Universal Routing Platforms Overview on page 567
- CoS Features and Limitations on MX Series Routers on page 568
- CoS Features of the Router Hardware, PIC, MIC, and MPC Interface Families on page 681
- How Schedulers Define Output Queue Properties on page 254
- *Subscriber Access Network Overview*
- *CoS for Subscriber Access Overview*
- *Hierarchical Class of Service for Subscriber Management Overview*

The Junos OS hierarchical schedulers support up to five levels of scheduler hierarchies on MX Series devices when using enhanced queuing Dense Port Concentrators (DPCs) or fine-grained queuing Modular Port Concentrators (MPCs), and Modular Interface Cards (MICs). It is important to know the capabilities of your hardware with respect to HCoS. The following are a few tips to help you:

- Only certain hardware supports the five-level scheduler hierarchy of HCoS.
- The number of queues and logical interfaces supported is dependent upon exactly what hardware you are using.
- The MX Series Packet Forwarding Engine handles guaranteed bandwidth and scheduler node weight differently than other Packet Forwarding Engines.
- The fine-grained queuing MPCs and MICs have a certain granularity with respect to the shaping and delay buffer values. The values used are not necessarily exactly the values configured.

To learn more about platform support for HCoS, use the Juniper Networks Feature Explorer (https://pathfinder.juniper.net/feature-explorer/). In the Feature Explorer, search on *hierarchical schedulers*.

Copyright © 2019, Juniper Networks, Inc.

In addition, it is important to note the following:

- HCoS is most frequently used to enforce service level agreements at the subscriber edged using dynamic traffic control profiles.
- Hierarchical schedulers can also be applied to Ethernet pseudowire interfaces, aggregated Ethernet interfaces, Layer 2 Tunnel Protocol (L2TP) network server (LNS) inline services, and GRE tunnels.
- Hierarchical ingress policing is a feature that is complimentary to and often used in conjunction with HCoS.
- There are other features in Junos OS that have similar sounding names.

NOTE: The *hierarchical scheduler and shaper* feature supported on the SRX Series devices is not the HCoS feature described here.

Before planning HCoS for you network, you should learn about HCoS, define you needs, plan how you want to implement HCoS, and test the operation in a simulated environment.

Table 42: Resources for Learning More About HCoS

Document	Description
Day One: Deploying Basic QoS Juniper Networks Books	This book is a good resource for learning the basics of CoS on Juniper Networks devices.
Juniper MX-Series O'Reilly Media	Learn about the advanced features of HCoS. This book provides an in-depth description of how HCoS works and how it can be deployed. It also provides a lab tested topology and configuration example.
Day One: Dynamic Subscriber Management Juniper Networks Books	Learn how to use HCoS in conjunction with dynamic traffic control profiles for subscriber management. This book also includes troubleshooting.
QoS Enabled Networks John Wiley & Sons	This book is an additional source for studying QoS.

Documentation related to HCoS is consolidated in the *Hierarchical Class of Service Feature Guide*.

Related Documentation

- *Hierarchical Class of Service for Subscriber Management Overview*
- Hierarchical Class of Service Network Scenarios on page 336
- Understanding Hierarchical Scheduling on page 337

Copyright © 2019, Juniper Networks, Inc.

Hierarchical Class of Service Network Scenarios

Hierarchical class of service (HCoS) can be used to provide granular control of traffic for a variety of different applications.

NOTE: Hierarchical class of service is also referred to as Hierarchical Quality of Service (HQoS) in other vendor's documentation.

Hierarchical class of service is most frequently used in the following scenarios:

Services to Subscribers

Multiservice network operators face a challenge to provide different types of services on the same infrastructure to residential and business subscribers. The network operator needs to make sure each subscriber gets the network resources they paid for and each service gets the network resources it needs to operate properly.

If no CoS is applied, one service could consume most of the bandwidth of the transmission infrastructure and starve the other services.

Using hierarchical class of service, the network edge device can have up to five levels of scheduling and prioritization. So the traffic can be shaped and prioritized per customer and per service type. Controlling traffic in this way provides the ability to deliver the required service level for each subscriber for each service type.

By allowing network operators to consolidate different services and multiple customers on the same physical infrastructure, hierarchical class of service helps maximize the ability to offer revenue generating services while simultaneously minimizing capital cost.

Services to Businesses

Hierarchical class of service is a valuable tool for service providers that support business customers who are running applications with different prioritization and scheduling requirements over the same infrastructure. In this scenario hierarchical class of service allows lower priority traffic to fully utilize the available bandwidth on a port, while simultaneously ensuring low latency and guaranteed bandwidth to higher priority traffic on the same port.

This allows a provider to consolidate different services on the same physical device and physical infrastructure thus optimizing network resources while maintaining the required level of service.

All of this maximizes revenue and minimizes cost

Wireless Backhaul

In a cellular network the operator might want to offer business services along with its cell tower traffic. One of the main challenges is to make sure that the time-sensitive cell traffic is not affected by the business services running on the same infrastructure. Each type of traffic has its own priority flows and bandwidth constraints. For example, wireless

Copyright © 2019, Juniper Networks, Inc.

backhaul is very sensitive to fluctuations in the packet stream (Jitter) because it relies on synchronization.

In this scenario, hierarchical class of service allows each type of traffic to receive the required resources and quality of service while being delivered over the same infrastructure.

By consolidate different services on the same physical infrastructure, HCoS helps maximize revenue and minimize cost.

Related Documentation

- Hierarchical Class of Service Overview on page 333
- *Hierarchical Class of Service for Subscriber Management Overview*

Understanding Hierarchical Scheduling

Hierarchical class of service (HCoS) is a set of capabilities that enable you to apply unique CoS treatment for network traffic based on criteria such as user, application, VLAN, and physical port.

This allows you to support the requirements of different services, applications, and users on the same physical device and physical infrastructure.

This topic covers the following information:

- Hierarchical Scheduling Terminology on page 337
- Scheduler Node-Level Designations in Hierarchical Scheduling on page 338
- Hierarchical Scheduling at Non-Leaf Nodes on page 339

Hierarchical Scheduling Terminology

Hierarchical scheduling introduces some new CoS terms and also uses some familiar terms in different contexts:

- Customer VLAN (C-VLAN)—A C-VLAN, defined by IEEE 802.1ad. A stacked VLAN contains an outer tag corresponding to the S-VLAN, and an inner tag corresponding to the C-VLAN. A C-VLAN often corresponds to CPE. Scheduling and shaping is often used on a C-VLAN to establish minimum and maximum bandwidth limits for a customer. See also *S-VLAN*.
- Interface set—A logical group of interfaces that describe the characteristics of set of service VLANs, logical interfaces, customer VLANs, or aggregated Ethernet interfaces. Interface sets establish the set and name the traffic control profiles. See also *Service VLAN*.
- Scheduler— A scheduler defines the scheduling and queuing characteristics of a queue. Transmit rate, scheduler priority, and buffer size can be specified. In addition, a drop profile may be referenced to describe WRED congestion control aspects of the queue. See also *Scheduler map*.

Copyright © 2019, Juniper Networks, Inc.

- Scheduler map—A scheduler map is referenced by traffic control profiles to define queues. The scheduler map establishes the queues that comprise a scheduler node and associates a forwarding class with a scheduler. See also *Scheduler*.
- Stacked VLAN—An encapsulation on an S-VLAN with an outer tag corresponding to the S-VLAN, and an inner tag corresponding to the C-VLAN. See also *Service VLAN* and *Customer VLAN*.
- Service VLAN (S-VLAN)—An S-VLAN, defined by IEEE 802.1ad, often corresponds to a network aggregation device such as a DSLAM. Scheduling and shaping is often established for an S-VLAN to provide CoS for downstream devices with little buffering and simple schedulers. See also *Customer VLAN*.
- Traffic control profile—Defines the characteristics of a scheduler node. Traffic control profiles are used at several levels of the CLI, including the physical interface, interface set, and logical interface levels. Scheduling and queuing characteristics can be defined for the scheduler node using the **shaping-rate**, **guaranteed-rate**, and **delay-buffer-rate** statements. Queues over these scheduler nodes are defined by referencing a scheduler map. See also *Scheduler* and *Scheduler map*.
- VLAN—Virtual LAN, defined on an Ethernet logical interface.

Scheduler Node-Level Designations in Hierarchical Scheduling

Scheduler hierarchies are composed of nodes and queues. Queues terminate the hierarchy. Nodes can be either root nodes, leaf nodes, or internal (non-leaf) nodes. Internal nodes are nodes that have other nodes as “children” in the hierarchy.

Scheduler hierarchies consist of levels, starting with Level 1 at the physical port. This chapter establishes a four-level scheduler hierarchy which, when fully configured, consists of the physical interface (Level 1), the interface set (Level 2), one or more logical interfaces (Level 3), and one or more queues (Level 4).

NOTE: **Beginning with Junos OS Release 16.1, certain MPCs on MX Series devices support up to five levels of scheduler hierarchies. The concepts presented in this topic apply similarly to five scheduler hierarchy levels.**

Table 43 on page 338 describes the possible combinations of scheduler nodes and their corresponding node level designations for a hierarchical queuing MIC or MPC.

Table 43: Node Levels Designations in Hierarchical Scheduling

Scheduler Configuration for Hierarchical CoS	Hierarchical CoS Scheduler Nodes			
	Root Node	Internal (Non-Leaf) Nodes		Leaf Node
	Level 1	Level 2	Level 3	Level 4
One or more traffic control profiles configured on logical interfaces, but no interface-sets configured	Physical interface	—	One or more logical interfaces	One or more queues

Copyright © 2019, Juniper Networks, Inc.

Table 43: Node Levels Designations in Hierarchical Scheduling (continued)

Scheduler Configuration for Hierarchical CoS	Hierarchical CoS Scheduler Nodes			
	Root Node	Internal (Non-Leaf) Nodes		Leaf Node
	Level 1	Level 2	Level 3	Level 4
Interface-sets (collections of logical interfaces) configured, but no traffic-control profiles configured on logical interfaces	Physical interface	—	Interface-set	One or more queues
Fully configured scheduler nodes	Physical interface	Interface-set	One or more logical interfaces	One or more queues

The table illustrates how the configuration of an interface set or logical interface affects the terminology of hierarchical scheduler nodes. For example, suppose you configure an **interface-set** statement with logical interfaces (such as **unit 0** and **unit 2**) and a queue. In this case, the interface-set is an internal node at Level 2 of the scheduler node hierarchy. However, if there are no traffic control profiles attached to logical interfaces, then the interface set is at Level 3 of the hierarchy.

Hierarchical Scheduling at Non-Leaf Nodes

Whereas standard CoS scheduling is based on the scheduling and queuing characteristics of a router's egress ports and their queues, hierarchical CoS scheduling is based on the scheduling and queuing characteristics that span a hierarchy of *scheduler nodes* over a port. The hierarchy begins at Level 1, a *root node* at the physical interface (port) level of the CLI hierarchy and terminates at Level 4, a *leaf node* at the queue level. Between the root and leaf nodes of any scheduler hierarchy are one or more *internal nodes*, which are non-root nodes that have other nodes as "children" in the hierarchy.

Whereas you configure standard CoS scheduling by applying a scheduler map to each egress port to specify a forwarding class and a queue priority level, you configure hierarchical CoS scheduling with additional parameters. To configure hierarchical CoS scheduling, you apply a scheduler map to the queue level (Level 4) of a scheduler hierarchy, and you can apply a different traffic control profile at each of the other levels. A traffic control profile specifies not only a scheduler map (forwarding class and queue priority level) but also optional shaping rate (PIR), guaranteed transmit rate (CIR), burst rate, delay buffer rate, and drop profile.

Copyright © 2019, Juniper Networks, Inc.

Release History Table

Release	Description
16.1	Beginning with Junos OS Release 16.1, certain MPCs on MX Series devices support up to five levels of scheduler hierarchies.

Priority Propagation in Hierarchical Scheduling

Priority propagation is performed for MX Series router output Interfaces on Enhanced Queuing DPCs, MICs, and MPCs, and for M Series and T Series router output interfaces on IQ2E PICs. Priority propagation is useful for mixed traffic environments when, for example, you want to make sure that the voice traffic of one customer does not suffer due to the data traffic of another customer. Nodes and queues are serviced in the order of their priority. The default priority of a queue is low, and you can explicitly configure a queue priority by including the priority statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level.

You cannot directly configure the priorities of all hierarchical scheduling elements. The priorities of internal nodes, for example, are determined as follows:

- The highest priority of an active child, that is, a child currently containing traffic. (Interface sets only take the highest priority of their active children.)
- Whether the node is above its configured guaranteed rate (CIR) or not (this is only relevant if the physical interface is in CIR mode).

Each queue has a configured priority and a hardware priority. The usual mapping between the configured priority and the hardware priority is shown in Table 44 on page 340.

Table 44: Queue Priority

Configured Priority	Hardware Priority
Strict-high	0
High	0
Medium-high	1
Medium-low	1
Low	2
MPCs also have configurable CLI priorities of excess-priority high, excess-priority medium-high, excess-priority medium-low, and excess-priority low. These priorities only take effect above the guaranteed rate.	

In CIR mode, the priority for each internal node depends on whether the highest active child node is above or below the guaranteed rate. The mapping between the highest active child's priority and the hardware priority below and above the guaranteed rate is shown in Table 45 on page 341.

Copyright © 2019, Juniper Networks, Inc.

Table 45: Internal Node Queue Priority for CIR Mode

Configured Priority of Highest Active Child Node	Hardware Priority Below Guaranteed Rate	Hardware Priority Above Guaranteed Rate
Strict-high	0	0
High	0	3
Medium-high	1	3
Medium-low	1	3
Low	2	3
Excess-priority high*	N/A	3
Excess-priority medium-high*	N/A	3
Excess-priority medium-low*	N/A	4
Excess-priority low*	N/A	4
* MPCs only		

In PIR-only mode, nodes cannot send if they are above the configured shaping rate. The mapping between the configured priority and the hardware priority is for PIR-only mode is shown in Table 46 on page 341.

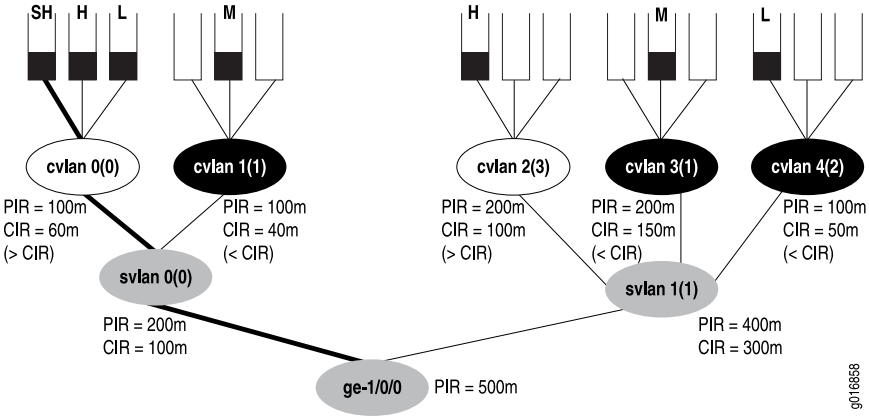
Table 46: Internal Node Queue Priority for PIR-Only Mode

Configured Priority	Hardware Priority
Strict-high	0
High	0
Medium-high	1
Medium-low	1
Low	2

A physical interface with hierarchical schedulers configured is shown in Figure 36 on page 342. The configured priorities are shown for each queue at the top of the figure. The hardware priorities for each node are shown in parentheses. Each node also shows any configured shaping rate (PIR) or guaranteed rate (CIR) and whether or not the queues is above or below the CIR. The nodes are shown in one of three states: above the CIR (clear), below the CIR (dark), or in a condition where the CIR does not matter (gray).

Copyright © 2019, Juniper Networks, Inc.

Figure 36: Hierarchical Schedulers and Priorities



In the figure, the strict-high queue for customer VLAN 0 (cvlan 0) receives service first, even though the customer VLAN is above the configured CIR (see Table 45 on page 341 for the reason: strict-high always has hardware priority 0 regardless of CIR state). Once that queue has been drained, and the priority of the node has become 3 instead of 0 (due to the lack of strict-high traffic), the system moves on to the medium queues next (cvlan 1 and cvlan 3), draining them in a round robin fashion (empty queue lose their hardware priority). The low queue on cvlan 4 (priority 2) is sent next, because that mode is below the CIR. Then the high queues on cvlan 0 and cvlan2 (both now with priority 3) are drained in a round robin fashion, and finally the low queue on cvlan 0 is drained (thanks to svlan 0 having a priority of 3).

Related Documentation

- CoS on Enhanced IQ2 PICs Overview on page 794
- Enhanced Queuing DPC CoS Properties on page 908
- CoS Features and Limitations on MIC and MPC Interfaces on page 930
- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972

Configuring Hierarchical Schedulers for CoS

In metro Ethernet environments, a virtual LAN (VLAN) typically corresponds to a customer premises equipment (CPE) device and the VLANs are identified by an inner VLAN tag on Ethernet frames (called the customer VLAN, or C-VLAN, tag). A set of VLANs can be grouped at the DSL access multiplexer (DSLAM) and identified by using the same outer VLAN tag (called the service VLAN, or S-VLAN, tag). The service VLANs are typically gathered at the Broadband Remote Access Server (B-RAS) level.Hierarchical schedulers let you provide shaping and scheduling at the service VLAN level as well as other levels, such as the physical interface. In other words, you can group a set of logical interfaces and then apply scheduling and shaping parameters to the logical interface set as well as to other levels.

On Juniper Networks MX Series 5G Universal Routing Platforms and systems with Enhanced IQ2 (IQ2E) PICs, you can apply CoS shaping and scheduling at one of four different levels, including the VLAN set level. You can only use this configuration on MX

Copyright © 2019, Juniper Networks, Inc.

Series routers or IQ2E PICs. Beginning with Junos OS Release 16.1, certain MPCs support up to five levels of scheduler hierarchies.

The supported scheduler hierarchy is as follows:

- The physical interface (level 1)
- The service VLAN (level 2 is unique to MX Series routers)
- The logical interface or customer VLAN (level 3)
- The queue (level 4)

Users can specify a traffic control profile (**output-traffic-control-profile** that can specify a shaping rate, a guaranteed rate, and a scheduler map with transmit rate and buffer delay. The scheduler map contains the mapping of queues (forwarding classes) to their respective schedulers (schedulers define the properties for the queue). Queue properties can specify a transmit rate and buffer management parameters such as buffer size and drop profile.

To configure CoS hierarchical scheduling, you must enable hierarchical scheduling by including the **hierarchical-scheduler** statement at the physical interface.

Related Documentation

- Understanding Hierarchical Scheduling on page 337
- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972
- CoS on Enhanced IQ2 PICs Overview on page 794
- *Understanding Hierarchical CoS for Subscriber Interfaces*

Hierarchical Schedulers and Traffic Control Profiles

When used, the interface set level of the hierarchy falls between the physical interface level (Level 1) and the logical interface (Level 3). Queues are always Level 4 of the hierarchy.

NOTE: **Beginning with Junos OS Release 16.1, certain MPCs on MX Series devices support up to five levels of scheduler hierarchies. The concepts presented in this topic apply similarly to five scheduler hierarchy levels.**

Hierarchical schedulers add CoS parameters to the interface-set level of the configuration. They use traffic control profiles to set values for parameters such as shaping rate (the peak information rate [PIR]), guaranteed rate (the committed information rate [CIR] on these interfaces), scheduler maps (assigning queues and resources to traffic), and so on.

The following CoS configuration places the following parameters in traffic control profiles at various levels:

- Traffic control profile at the port level (**tcp-port-level1**):

Copyright © 2019, Juniper Networks, Inc.

  - A shaping rate (PIR) of 100 Mbps
  - A delay buffer rate of 100 Mbps
- Traffic control profile at the interface set level (**tcp-interface-level2**):
  - A shaping rate (PIR) of 60 Mbps
  - A guaranteed rate (CIR) of 40 Mbps
- Traffic control profile at the logical interface level (**tcp-unit-level3**):
  - A shaping rate (PIR) of 50 Mbps
  - A guaranteed rate (CIR) of 30 Mbps
  - A scheduler map called **smap1** to hold various queue properties (level 4)
  - A delay buffer rate of 40 Mbps

In this case, the traffic control profiles look like this:

```
[edit class-of-service traffic-control-profiles]
tcp-port-level1 { # This is the physical port level
   shaping-rate 100m;
   delay-buffer-rate 100m;
}
tcp-interface-level2 { # This is the interface set level
   shaping-rate 60m;
   guaranteed-rate 40m;
}
tcp-unit-level3 { # This is the logical interface level
   shaping-rate 50m;
   guaranteed-rate 30m;
   scheduler-map smap1;
   delay-buffer-rate 40m;
}
```

Once configured, the traffic control profiles must be applied to the proper places in the CoS interfaces hierarchy.

```
[edit class-of-service interfaces]
interface-set level-2 {
   output-traffic-control-profile tcp-interface-level-2;
}
ge-0/1/0 {
   output-traffic-control-profile tcp-port-level-1;
   unit 0 {
      output-traffic-control-profile tcp-unit-level-3;
   }
}
```

In all cases, the properties for level 4 of the hierarchical schedulers are determined by the scheduler map.

Copyright © 2019, Juniper Networks, Inc.

Release History Table

Release	Description
16.1	Beginning with Junos OS Release 16.1, certain MPCs on MX Series devices support up to five levels of scheduler hierarchies.

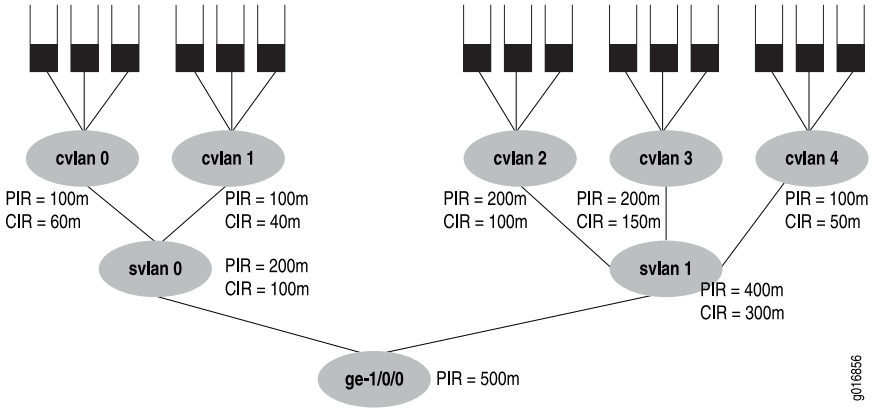
Related Documentation

- Oversubscribing Interface Bandwidth on page 273
- Providing a Guaranteed Minimum Rate on page 285
- Configuring Scheduler Maps on page 258
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

Example: Building a Four-Level Hierarchy of Schedulers

This section provides a more complete example of building a 4-level hierarchy of schedulers. The configuration parameters are shown in Figure 37 on page 345. The queues are shown at the top of the figure with the other three levels of the hierarchy below.

Figure 37: Building a Scheduler Hierarchy



The figure's PIR values are configured as the shaping rates and the CIRs are configured as the guaranteed rate on the Ethernet interface **ge-1/0/0**. The PIR can be oversubscribed (that is, the sum of the children PIRs can exceed the parent's, as in **svlan 1**, where 200 + 200 + 100 exceeds the parent rate of 400)). However, the sum of the children node level's CIRs must never exceed the parent node's CIR, as shown in all the service VLANs (otherwise, the guaranteed rate could never be provided in all cases).

This configuration example presents all details of the CoS configuration for the interface in the figure (**ge-1/0/0**), including:

- Configuring the Interface Sets on page 346
- Configuring the Interfaces on page 346
- Configuring the Traffic Control Profiles on page 347
- Configuring the Schedulers on page 347

Copyright © 2019, Juniper Networks, Inc.

- Configuring the Drop Profiles on page 348
- Configuring the Scheduler Maps on page 348
- Applying the Traffic Control Profiles on page 349

Configuring the Interface Sets

```
[edit interfaces]
interface-set svlan-0 {
   interface ge-1/0/0 {
      unit 0;
      unit 1;
   }
}
interface-set svlan-1 {
   interface ge-1/0/0 {
      unit 2;
      unit 3;
      unit 4;
   }
}
```

Configuring the Interfaces

The keyword to configure hierarchical schedulers is at the physical interface level, as is VLAN tagging and the VLAN IDs. In this example, the interface sets are defined by logical interfaces (units) and not outer VLAN tags. All VLAN tags in this example are customer VLAN tags.

```
[edit interface ge-1/0/0]
hierarchical-scheduler;
vlan-tagging;
unit 0 {
   vlan-id 100;
}
unit 1 {
   vlan-id 101;
}
unit 2 {
   vlan-id 102;
}
unit 3 {
   vlan-id 103;
}
unit 4 {
   vlan-id 104;
}
```


Copyright © 2019, Juniper Networks, Inc.

Configuring the Traffic Control Profiles

The traffic control profiles hold parameters for levels above the queue level of the scheduler hierarchy. This section defines traffic control profiles for both the service VLAN level (logical interfaces) and the customer VLAN (VLAN tag) level.

```
[edit class-of-service traffic-control-profiles]
tcp-500m-shaping-rate {
   shaping-rate 500m;
}
tcp-svlan0 {
   shaping-rate 200m;
   guaranteed-rate 100m;
   delay-buffer-rate 300m; # This parameter is not shown in the figure.
}
tcp-svlan1 {
   shaping-rate 400m;
   guaranteed-rate 300m;
   delay-buffer-rate 100m; # This parameter is not shown in the figure.
}
tcp-cvlan0 {
   shaping-rate 100m;
   guaranteed-rate 60m;
   scheduler-map tcp-map-cvlan0; # Applies scheduler maps to customer VLANs.
}
tcp-cvlan1 {
   shaping-rate 100m;
   guaranteed-rate 40m;
   scheduler-map tcp-map-cvlan1; # Applies scheduler maps to customer VLANs.
}
tcp-cvlan2 {
   shaping-rate 200m;
   guaranteed-rate 100m;
   scheduler-map tcp-map-cvlanx; # Applies scheduler maps to customer VLANs.
}
tcp-cvlan3 {
   shaping-rate 200m;
   guaranteed-rate 150m;
   scheduler-map tcp-map-cvlanx; # Applies scheduler maps to customer VLANs
}
tcp-cvlan4 {
   shaping-rate 100m;
   guaranteed-rate 50m;
   scheduler-map tcp-map-cvlanx; # Applies scheduler maps to customer VLANs
}
```

Configuring the Schedulers

The schedulers hold the information about the queues, the last level of the hierarchy. Note the consistent naming schemes applied to repetitive elements in all parts of this example.

```
[edit class-of-service schedulers]
sched-cvlan0-qx {
```

Copyright © 2019, Juniper Networks, Inc.

```
    priority low;
    transmit-rate 20m;
    buffer-size temporal 100ms;
    drop-profile loss-priority low dp-low;
    drop-profile loss-priority high dp-high;
}
sched-cvlan1-q0 {
    priority high;
    transmit-rate 20m;
    buffer-size percent 40;
    drop-profile loss-priority low dp-low;
    drop-profile loss-priority high dp-high;
}
sched-cvlanx-qx {
    transmit-rate percent 30;
    buffer-size percent 30;
    drop-profile loss-priority low dp-low;
    drop-profile loss-priority high dp-high;
}
sched-cvlan1-qx {
    transmit-rate 10m;
    buffer-size temporal 100ms;
    drop-profile loss-priority low dp-low;
    drop-profile loss-priority high dp-high;
}
```

Configuring the Drop Profiles

This section configures the drop profiles for the example. For more information about interpolated drop profiles, see "Managing Congestion Using RED Drop Profiles and Packet Loss Priorities" on page 351.

```
[edit class-of-service drop-profiles]
dp-low {
    interpolate fill-level 80 drop-probability 80;
    interpolate fill-level 100 drop-probability 100;
}
dp-high {
    interpolate fill-level 60 drop-probability 80;
    interpolate fill-level 80 drop-probability 100;
}
```

Configuring the Scheduler Maps

This section configures the scheduler maps for the example. Each one references a scheduler configured in "Configuring the Schedulers" on page 347.

```
[edit class-of-service scheduler-maps]
tcp-map-cvlan0 {
    forwarding-class voice scheduler sched-cvlan0-qx;
    forwarding-class video scheduler sched-cvlan0-qx;
    forwarding-class data scheduler sched-cvlan0-qx;
}
tcp-map-cvlan1 {
```


Copyright © 2019, Juniper Networks, Inc.

```
    forwarding-class voice scheduler sched-cvlan1-q0;
    forwarding-class video scheduler sched-cvlan1-qx;
    forwarding-class data scheduler sched-cvlan1-qx;
}
tcp-map-cvlanx {
    forwarding-class voice scheduler sched-cvlanx-qx;
    forwarding-class video scheduler sched-cvlanx-qx;
    forwarding-class data scheduler sched-cvlanx-qx;
}
```

Applying the Traffic Control Profiles

This section applies the traffic control profiles to the proper levels of the hierarchy.

NOTE: Although a shaping rate can be applied directly to the physical interface, hierarchical schedulers must use a traffic control profile to hold this parameter.

```
[edit class-of-service interfaces]
ge-1/0/0 {
    output-traffic-control-profile tcp-500m-shaping-rate;
    unit 0 {
        output-traffic-control-profile tcp-cvlan0;
    }
    unit 1 {
        output-traffic-control-profile tcp-cvlan1;
    }
    unit 2 {
        output-traffic-control-profile tcp-cvlan2;
    }
    unit 3 {
        output-traffic-control-profile tcp-cvlan3;
    }
    unit 4 {
        output-traffic-control-profile tcp-cvlan4;
    }
}
interface-set svlan0 {
    output-traffic-control-profile tcp-svlan0;
}
interface-set svlan1 {
    output-traffic-control-profile tcp-svlan1;
}
```

Copyright © 2019, Juniper Networks, Inc.

NOTE: You should be careful when using a **show interfaces queue** command that references nonexistent class-of-service logical interfaces. When multiple logical interfaces (units) are not configured under the same interface set or physical interface, but are referenced by a command such as **show interfaces queue ge-10/0/1.12 forwarding-class be** or **show interfaces queue ge-10/0/1.13 forwarding-class be** (where logical units 12 and 13 are not configured as a class-of-service interfaces), these interfaces display the same traffic statistics for each logical interface. In other words, even if there is no traffic passing through a particular unconfigured logical interface, as long as one or more of the other unconfigured logical interfaces under the same interface set or physical interface is passing traffic, this particular logical interface displays statistics counters showing the total amount of traffic passed through all other unconfigured logical interfaces together.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 9

Controlling Congestion with Scheduler RED Drop Profiles and Buffers

- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351
- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355
- Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 359
- Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 361
- Mapping PLP to RED Drop Profiles on page 363
- Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size on page 365
- Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 381
- Example: Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 383

Managing Congestion Using RED Drop Profiles and Packet Loss Priorities

You can configure two parameters to control congestion at the output stage. The first parameter defines the *delay-buffer bandwidth*, which provides packet buffer space to absorb burst traffic up to the specified duration of delay. Once the specified delay buffer becomes full, packets with 100 percent drop probability are dropped from the head of the buffer. For more information, see "Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size" on page 365.

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

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

Copyright © 2019, Juniper Networks, Inc.

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

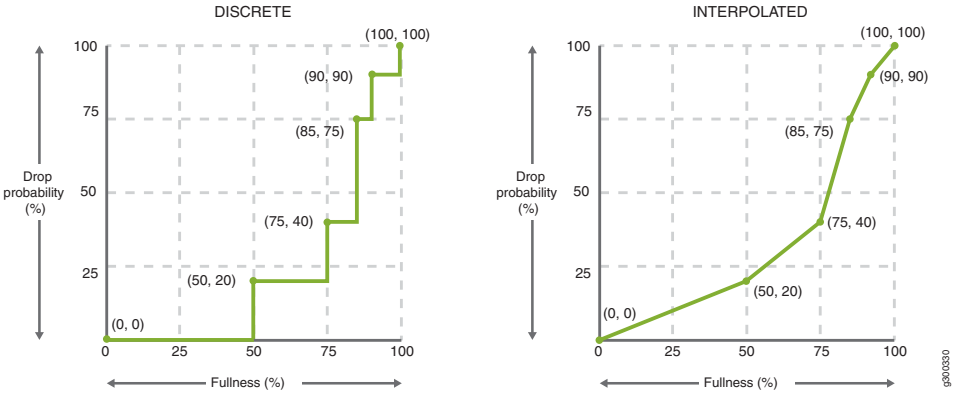
When you configure drop profiles, there are two important values: the queue fullness and the drop probability. The *queue fullness* represents a percentage of the memory used to store packets in relation to the total amount that has been allocated for that specific queue. Similarly, the *drop probability* is a percentage value that correlates to the likelihood that an individual packet is dropped from the network. How these two variables function is illustrated in graph format, as shown in Figure 38 on page 352.

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

- Physical or logical interfaces hosted on MICs in Queuing or Enhanced Queuing MPCs for MX Series routers support up to 64 (fill level, drop probability) pairs per discrete or interpolated drop profile.
- Physical or logical interfaces hosted on Enhanced Queuing DPCs for MX Series routers support up to 64 (fill level, drop probability) pairs per discrete drop profile or 2 pairs per interpolated drop profile. For more information, see "Configuring WRED on Enhanced Queuing DPCs" on page 912.
- Physical or logical interfaces hosted on IQ2 PICs or IQE PICs support up to two (fill level, drop probability) pairs per discrete or interpolated drop profile.

Figure 38 on page 352 shows both a discrete and an interpolated graph. Although the formation of these graph lines is different, the application of the profile is the same. When a packet reaches the head of the queue, a random number between 0 and 100 is calculated by the router or switch. This random number is plotted against the drop profile using the current queue fullness of that particular queue. When the random number falls above the graph line, the packet is transmitted onto the physical media. When the number falls below the graph line, the packet is dropped from the network.

Figure 38: Discrete and Interpolated Drop Profiles



Drop profiles are created by defining multiple fill levels and drop probabilities and can be illustrated by graphs in which the x axis represents the fill level and the y axis represents the drop probability.

Copyright © 2019, Juniper Networks, Inc.

To create the discrete profile graph as shown in Figure 38 on page 352 on the left, the software begins at the bottom-left corner, representing a 0-percent fill level and a 0-percent drop probability. This configuration creates a line horizontally to the right on the fullness level (l) x-axis until it reaches the first defined fill level, 50-percent for this configuration, which is designated to have a drop probability (p) of 20-percent. The software then continues the line horizontally along the fill level until the next drop probability is reached at the designated data point of 75-percent fill level, which has a designated drop-probability of 40-percent. The line is then continued horizontally to the next fill level of 85-percent and the designated drop probability of 75-percent. The line continues horizontally to the next designated fill level of 90-percent, which has a designated drop probability of 90-percent, and a line is created to data point 90-percent (l), 90-percent (p) (l90 p90). From the l90 p90 point, the line continues horizontally to the 100-percent fill level, which has a drop probability of 100 percent, at which the line rises to the end-point of 100-100, which is 100 percent fill level with a 100 percent drop probability.

If an interpolated drop profile is specified, in the first quadrant the initial line segment spans from the origin (0,0) to the next defined point. From that defined fill-level/drop-probability point, a second line runs to the next point, and so forth, until a final line segment connects (100, 100). The software automatically constructs a drop profile containing 64 fill levels at drop probabilities that approximate the calculated line segments.

You can create a smoother graph line by configuring the profile with the **interpolate** statement. This enables the software to automatically generate 64 data points on the graph beginning at (0, 0) and ending at (100, 100). Along the way, the graph line intersects specific data points that you have defined.

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

Loss priorities allow you to set the priority of dropping a packet. Loss priority affects the scheduling of a packet without affecting the packet's relative ordering. You can use the packet loss priority (PLP) bit as part of a congestion control strategy. You can use the loss priority setting to identify packets that have experienced congestion. Typically you mark packets exceeding some service level with a high loss priority. You set loss priority by configuring a classifier or a policer. The loss priority is used later in the workflow to select one of the drop profiles used by RED.

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

You can configure a maximum of 32 different drop profiles.

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

Copyright © 2019, Juniper Networks, Inc.

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

If you configure no drop profiles on Juniper Networks M320 Multiservice Edge Routers or T Series Core Routers, random early detection (RED) is in effect by default and functions as the primary mechanism for managing congestion. In the default RED drop profile, when the fill-level is 0 percent, the drop probability is 0 percent. When the fill-level is 100 percent, the drop probability is 100 percent.

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

Related Documentation

- drop-probability (Interpolated Value) on page 1083
- drop-probability (Percentage) on page 1084

Copyright © 2019, Juniper Networks, Inc.

Defining Packet Drop Behavior by Configuring RED Drop Profiles

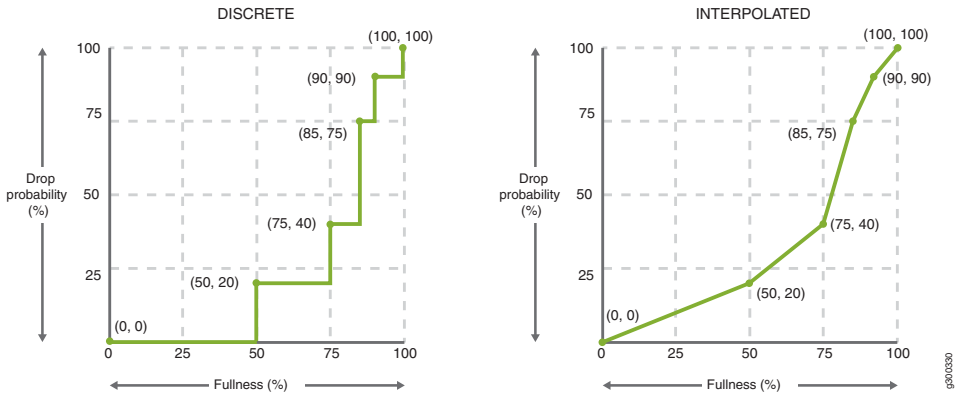
You enable *random early detection (RED)* by applying a drop profile to a scheduler. When RED is operational on an interface, the queue no longer drops packets from the tail of the queue. Rather, packets are dropped after they reach the head of the queue.

Copyright © 2019, Juniper Networks, Inc.

To configure a drop profile, include either the **interpolate** statement and its options, or the fill-level and drop-probability ***percentage*** values. These two alternatives enable you to configure either each drop probability at up to 64 fill-level/drop-probability paired values, or a profile represented as a series of line segments, as discussed in "Managing Congestion Using RED Drop Profiles and Packet Loss Priorities" on page 351.

For example, the following shows a discrete configuration and an interpolated configuration that correspond to the graphs in Figure 38 on page 352. The values defined in the configurations are matched to represent the data points in the graph lines.

Figure 39: Discrete and Interpolated Drop Profiles



Creating a Discrete Configuration

```
class-of-service {
  drop-profiles {
    discrete-style-profile {
      fill-level 0 drop-probability 0;
      fill-level 50 drop-probability 20;
      fill-level 75 drop-probability 40;
      fill-level 85 drop-probability 75;
      fill-level 90 drop-probability 90;
      fill-level 100 drop-probability 100;
    }
  }
}
```

To create the discrete profile graph as shown in Figure 38 on page 352 on the left, the software begins at the bottom-left corner, representing a 0-percent fill level and a 0-percent drop probability. This configuration creates a line horizontally to the right on the fullness level (l) until it reaches the first defined fill level, 50-percent for this configuration, which is designated to have a drop probability (p) of 20-percent. The software then continues the line horizontally along the fill level until the next drop probability is reached at the designated data point of 75-percent fill level, which has a designated drop-probability of 40-percent. The line is then continued horizontally to the next fill level of 85-percent and the designated drop probability of 75-percent. The line continues horizontally to the next designated fill level of 90-percent, which has a designated drop probability of 90-percent, and a line is created to data point 90-percent (l), 90-percent (p) (l90 p90). From the l90 p90 point, the line continues horizontally to

Copyright © 2019, Juniper Networks, Inc.

the 100-percent fill level, which has a drop probability of 100 percent, at which the line rises to the end-point of 100-100, which is 100 percent fill level with a 100 percent drop probability.

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

Creating an Interpolated Configuration

```
class-of-service {
  drop-profiles {
    interpolated-style-profile {
      interpolate {
        fill-level [ 0 50 75 85 90 100 ];
        drop-probability [ 0 20 40 75 90 100 ];
      }
    }
  }
}
```

To configure a drop profile:

1. Create the drop profile by specifying a name for it.

   ```
   [edit]
   user@host# edit class-of-service drop-profiles profile-name
   ```

2. (Optional) Specify the fill-level and drop-probability values for the drop profile.

   ```
   [edit class-of-service drop-profiles profile-name]
   user@host# set fill-level percentage drop-probability percentage
   ```

   Repeat this step for each fill-level and drop-probability.

3. (Optional) Specify values for interpolating the relationship between queue fill level and drop probability.

   ```
   [edit class-of-service drop-profiles profile-name]
   user@host# set interpolate drop-probability percentage drop-probability percentage
   ```

4. Verify your configuration.

   ```
   [edit class-of-service drop-profiles]
   user@host# show
   ```

5. Save your configuration.

   ```
   [edit class-of-service drop-profiles]
   user@host# commit
   ```

Copyright © 2019, Juniper Networks, Inc.

NOTE: After you configure a drop profile, you must assign the drop profile to a drop-profile map, and assign the drop-profile map to a scheduler, as discussed in "Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers" on page 359.

Related Documentation

- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351

Copyright © 2019, Juniper Networks, Inc.

Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers

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

Drop-profile maps associate RED drop profiles with a scheduler. The map examines the current loss priority setting of the packet (**low**, **medium-low**, **medium-high**, or **high**) and assigns a drop profile according to these values. For example, you can specify that all TCP packets with **low** loss priority are assigned a drop profile that you name **low-drop**. You can associate multiple drop-profile maps with a single queue.

The scheduler drop profile defines the drop probabilities across the range of delay-buffer occupancy, thereby supporting the RED process. Depending on the drop probabilities, RED might drop packets aggressively long before the buffer becomes full, or it might drop only a few packets even if the buffer is almost full. For information on how to configure drop profiles, see "Defining Packet Drop Behavior by Configuring RED Drop Profiles" on page 355.

By default, the drop profile is mapped to packets with low PLP and any protocol type.

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

The map sets the drop profile for a specific PLP and protocol type. The inputs for the map are the PLP and the protocol type. The output is the drop profile. In other words, the map sets the drop profile for each packet with a specific PLP and protocol type exiting the interface. For more information about how CoS maps work, see "Mapping CoS Component Inputs to Outputs" on page 11.

NOTE: On Juniper Network MX Series 5G Universal Routing Platforms, T4000 Core Routers, EX Series switches, and PTX Series Packet Transport Routers, you can configure only the **any** option for the **protocol** statement.

For each scheduler, you can configure separate drop profile maps for each loss priority.

You can configure a maximum of 32 different drop profiles.

In the following sample configuration, the **dp** drop profile is assigned to all packets exiting the interface with a medium-low PLP and belonging to any protocol:To configure this drop profile map:

1. Specify the name of the scheduler.

```
[edit]
user@host# edit class-of-service schedulers af
```

2. Define the loss-priority value for a drop profile, the protocol type, and the name of the drop profile..

```
[[edit class-of-service schedulers af]
user@host# set drop-profile-map loss-priority medium-low protocol any drop-profile
  dp
```

3. Verify your configuration.

```
[edit class-of-service]
user@host# show schedulers af
```

```
drop-profile-map loss-priority medium-low protocol any drop-profile dp;
```

4. Save your configuration.

```
[edit class-of-service]
user@host# commit
```

NOTE: To use this drop-profile map, you must configure the settings for the dp drop profile at the [edit class-of-service drop-profiles dp] hierarchy level..

Related Documentation

- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355
- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351

Copyright © 2019, Juniper Networks, Inc.

Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows

By default, the least significant bit of the CoS value sets the packet loss priority (PLP) value. For example, CoS value 000 is associated with PLP **low**, and CoS value 001 is associated with PLP **high**. In general, you can change the PLP by configuring a behavior aggregate (BA) or multifield classifier, as discussed in "Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic" on page 36 and "Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields" on page 97.

However, on Juniper Networks M320 Multiservice Edge Routers, MX Series 5G Universal Routing Platforms, and T Series Core Routers and EX Series switches that do not have tricolor marking enabled, the loss priority can be configured by setting the PLP within a multifield classifier or by behavior aggregate (BA) classifier. This setting can then be used by the appropriate drop profile map and rewrite rule.

On M320 routers and T Series routers with Enhanced II Flexible PIC Concentrators (FPCs) and tricolor marking enabled, you can set the PLP with a BA or multifield classifier, as described in "Configuring Behavior Aggregate Classifiers" on page 51 and "Using Multifield Classifiers to Set Packet Loss Priority" on page 101.

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

The following example shows a two-step procedure to override the default PLP settings on M320 routers.

The first part of this example specifies that while the DSCP code points are 110, the loss priority is set to **high**; however, on M320 routers, overriding the default PLP this way has no effect.

1. Configure the classifier name and specify it as type as DSCP.

```
[edit]
user@host# edit class-of-service classifiers dscp ba-classifier
```

2. Specify the forwarding class

```
[edit class-of-service classifiers dscp ba-classifier]
user@host# set forwarding-class expedited-forwarding loss-priority high code-points
  110
```

For M320 routers, use the following procedure to configure a multifield classifier that sets the PLP.

1. Under the **firewall** statement, specify a name for the filter.

```
edit
user@host# edit firewall filter ef-filter
```

Copyright © 2019, Juniper Networks, Inc.

2. Specify the term name and match criteria you want to look for in incoming packets.

```
[edit firewall filter ef-filter]
user@host# set term ef-multifield from precedence 6
```

3. Specify the action you want to take when a packet matches the conditions.

```
[edit firewall filter ef-filter]
user@host# set term ef-multifield then loss-priority high forwarding-class
  expedited-forwarding
```

4. Verify your configuration.

```
[edit firewall]
user@host# show
```

```
filter ef-filter {
    term ef-multifield {
        from {
            precedence 6;
        }
        then {
            loss-priority high;
            forwarding-class expeditd-forwarding;
        }
    }
}
```

5. Save your configuration.

```
[edit firewall]
user@host# commit
```

Related Documentation

- Mapping PLP to RED Drop Profiles on page 363
- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355
- Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 359
- Configuring Schedulers on page 258

Copyright © 2019, Juniper Networks, Inc.

Mapping PLP to RED Drop Profiles

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

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

Obviously, *low*, *medium-low*, *medium-high*, and *high* are relative terms, which by themselves have no meaning. Drop profiles define the meanings of the loss priorities. In the following example, the **low-drop** drop profile defines the meaning of **low** PLP as a 10 percent drop probability when the fill level is 75 percent and a 40 percent drop probability when the fill level is 95 percent. The **high-drop** drop profile defines the meaning of **high** PLP as a 50 percent drop probability when the fill level is 25 percent and a 90 percent drop probability when the fill level is 50 percent.

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

1. Create the low drop profile.

```
[edit]
user@host# edit class-of-service drop-profiles low-drop
```

2. Specify values for interpolating the relationship between the queue fill level and drop probability for the low drop profile.

```
[edit class-of-service drop-profiles low-drop]
user@host# edit interpolate
user@host# set drop-probability [10 40]
user@host# set fill-level [75 95]
```

3. Crate the high drop profile.

```
[edit class-of-service drop-profiles]
user@host# edit high-drop
```

4. Specify values for interpolating the relationship between the queue fill level and drop probability for the high drop profile.

```
[edit class-of-service drop-profiles high-drop]
user@host# edit interpolate
user@host# set drop-probability [50 90]
```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set fill-level [25 50]
```

5. Specify the scheduler name.

```
[edit class-of-service]
user@host# edit schedulers best effort
```

6. Define the loss-priority for each low drop profile.

```
[edit class-of-service schedulers best-effort]
user@host# set drop-profile-map loss-priority low protocol any drop-profile low-drop
```

7. Define the loss-priority for each high drop profile.

```
[edit class-of-service schedulers best-effort]
user@host# set drop-profile-map loss-priority high protocol any drop-profile high-drop
```

8. Verify your configuration.

```
[edit class-of-service]
user@host# show
```

```
drop-profiles {
    low-drop {
        interpolate {
            fill-level [ 75 95 ];
            drop-probability [ 10 40 ];
        }
    }
    high-drop {
        interpolate {
            fill-level [ 25 50 ];
            drop-probability [ 50 90 ];
        }
    }
}
```

```
schedulers {
    best-effort {
       drop-profile-map loss-priority low protocol any drop-profile low-drop;

      drop-profile-map loss-priority high protocol any drop-profile high-drop;

    }
}
```

9. Save your configuration.

```
[edit class-of-service]
user@host# commit
```


Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 361
- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355
- Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 359
- Configuring Schedulers on page 258

Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size

To control congestion at the output stage, you can configure the delay-buffer bandwidth. The delay-buffer bandwidth provides packet buffer space to absorb burst traffic up to the specified duration of delay. Once the specified delay buffer becomes full, packets with 100 percent drop probability are dropped from the head of the buffer.

The default scheduler transmission rate for queues 0 through 7 are 95, 0, 0, 0, 0, 0, 0, and 5 percent of the total available bandwidth.

The default buffer size percentages for queues 0 through 7 are 95, 0, 0, 0, 0, 0, 0, and 5 percent of the total available buffer. The total available buffer per queue differs by PIC type.

To configure the buffer size, include the **buffer-size** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level:

```
[edit class-of-service schedulers scheduler-name]
buffer-size (percent percentage | remainder | shared | temporal microseconds);
```

For each scheduler, you can configure the buffer size as one of the following:

- A percentage of the total buffer. The total buffer per queue is based on microseconds and differs by routing device type, as shown in Table 47 on page 366.
- The remaining buffer available. The remainder is the buffer percentage that is not assigned to other queues. For example, if you assign 40 percent of the delay buffer to queue 0, allow queue 3 to keep the default allotment of 5 percent, and assign the remainder to queue 7, then queue 7 uses approximately 55 percent of the delay buffer.
- Shared from the interface's buffer pool. On PTX Series routers, set a queue's buffer to be up to 100 percent of the interface's buffer. This option allows the queue's buffer to grow as large as 100 percent of the interface's buffer if and only if it is the only active queue for the interface.
- A temporal value, in microseconds. For the temporal setting, the queuing algorithm starts dropping packets when it queues more than a computed number of bytes. This maximum is computed by multiplying the transmission rate of the queue by the configured temporal value. The buffer size temporal value per queue differs by routing device type, as shown in Table 47 on page 366. The maximums apply to the logical interface, not each queue.

Copyright © 2019, Juniper Networks, Inc.

NOTE: In general, the default temporal buffer value is inversely related to the speed, or shaping rate, of the interface. As the speed of the interface increases, the interface needs less and less buffer to hold data, as it is possible for the interface to send more and more data.

Table 47: Buffer Size Temporal Value Ranges by Routing Device Type

Routing Devices	Temporal Value Ranges
M320 and T Series router FPCs, Type 1 and Type 2	1 through 80,000 microseconds
M320 and T Series router FPCs, Type 3. All ES cards (Type 1, 2, 3, and 4).	1 through 50,000 microseconds For PICs with greater than 40 Gbps of total bandwidth, the maximum temporal buffer size that can be configured for a scheduler is 40,000 microseconds instead of 50,000 microseconds.
M120 router FEBs and MX Series router nonenhanced Queuing DPCs, and EX Series switches	1 through 100,000 microseconds
M5, M7i, M10, and M10i router FPCs	1 through 100,000 microseconds
Other M Series router FPCs	1 through 200,000 microseconds
PTX Series Packet Transport Routers	1 through 100,000 microseconds
IQ PICs on all routers	1 through 100,000 microseconds
With Large Buffer Sizes Enabled	
IQ PICs on all routers	1 through 500,000 microseconds
Gigabit Ethernet IQ VLANs	
With shaping rate up to 10 Mbps	1 through 400,000 microseconds
With shaping rate up to 20 Mbps	1 through 300,000 microseconds
With shaping rate up to 30 Mbps	1 through 200,000 microseconds
With shaping rate up to 40 Mbps	1 through 150,000 microseconds
With shaping rate above 40 Mbps	1 through 100,000 microseconds

For more information about configuring delay buffers, see the following subtopics:

- Configuring Large Delay Buffers for Slower Interfaces on page 367
- Configuring the Maximum Delay Buffer for NxDS0 Interfaces on page 371

Copyright © 2019, Juniper Networks, Inc.

- Example: Configuring Large Delay Buffers for Slower Interfaces on page 374
- Example: Configuring the Delay Buffer Value for a Scheduler on page 375
- Example: Configuring the Physical Interface Shaping Rate on page 376
- Complete Configuration on page 377
- Enabling and Disabling the Memory Allocation Dynamic per Queue on page 378

Configuring Large Delay Buffers for Slower Interfaces

By default, T1, E1, and *N*xDS0 interfaces and DLCIs configured on channelized IQ PICs are limited to 100,000 microseconds of delay buffer. (The default average packet size on the IQ PIC is 40 bytes.) For these interfaces, it might be necessary to configure a larger buffer size to prevent congestion and packet dropping. You can do so on the following PICs:

- Channelized IQ
- 4-port E3 IQ
- Gigabit Ethernet IQ and IQ2

Congestion and packet dropping occur when large bursts of traffic are received by slower interfaces. This happens when faster interfaces pass traffic to slower interfaces, which is often the case when edge devices receive traffic from the core of the network. For example, a 100,000-microsecond T1 delay buffer can absorb only 20 percent of a 5000-microsecond burst of traffic from an upstream OC3 interface. In this case, 80 percent of the burst traffic is dropped.

Table 48 on page 367 shows some recommended buffer sizes needed to absorb typical burst sizes from various upstream interface types.

Table 48: Recommended Delay Buffer Sizes

Length of Burst	Upstream Interface	Downstream Interface	Recommended Buffer on Downstream Interface
5000 microseconds	OC3	E1 or T1	500,000 microseconds
5000 microseconds	E1 or T1	E1 or T1	100,000 microseconds
1000 microseconds	T3	E1 or T1	100,000 microseconds

To ensure that traffic is queued and transmitted properly on E1, T1, and *N*xDS0 interfaces and DLCIs, you can configure a buffer size larger than the default maximum. To enable larger buffer sizes to be configured:

1. Include the **q-pic-large-buffer (large-scale | small-scale)** statement at the **[edit chassis fpc *slot-number* pic *pic-number*]** hierarchy level.

```
[edit]
user@host# edit chassis fpc slot-number pic pic-number
user@host# set q-pic-large-buffer large-scale
```

Copyright © 2019, Juniper Networks, Inc.

If you specify the **large-scale** option, the feature supports a larger number of interfaces. If you specify **small-scale**, the default, then the feature supports a smaller number of interfaces.

When you include the **q-pic-large-buffer** statement in the configuration, the larger buffer is transparently available for allocation to scheduler queues. The larger buffer maximum varies by interface type, as shown in Table 49 on page 368.

Table 49: Maximum Delay Buffer with q-pic-large-buffer Enabled by Interface

Platform, PIC, or Interface Type	Maximum Buffer Size
With Large Buffer Sizes Not Enabled	
M320 and T Series router FPCs, Type 1 and Type 2	80,000 microseconds
M320 and T Series router FPCs, Type 3	50,000 microseconds
Other M Series router FPCs	200,000 microseconds
IQ PICs on all routers	100,000 microseconds
With Large Buffer Sizes Enabled	
Channelized T3 and channelized OC3 DLCIs—Maximum sizes vary by shaping rate:	
With shaping rate from 64,000 through 255,999 bps	4,000,000 microseconds
With shaping rate from 256,000 through 511,999 bps	2,000,000 microseconds
With shaping rate from 512,000 through 1,023,999 bps	1,000,000 microseconds
With shaping rate from 1,024,000 through 2,048,000 bps	500,000 microseconds
With shaping rate from 2,048,001 bps through 10 Mbps	400,000 microseconds
With shaping rate from 10,000,001 bps through 20 Mbps	300,000 microseconds
With shaping rate from 20,000,001 bps through 30 Mbps	200,000 microseconds
With shaping rate from 30,000,001 bps through 40 Mbps	150,000 microseconds
With shaping rate from 40,000,001 bps and above	100,000 microseconds
*N*xDS0 IQ Interfaces—Maximum sizes vary by channel size:	
1xDS0 through 3xDS0	4,000,000 microseconds
4xDS0 through 7xDS0	2,000,000 microseconds
8xDS0 through 15xDS0	1,000,000 microseconds
16xDS0 through 32xDS0	500,000 microseconds

Copyright © 2019, Juniper Networks, Inc.

Table 49: Maximum Delay Buffer with q-pic-large-buffer Enabled by Interface (continued)

Platform, PIC, or Interface Type	Maximum Buffer Size
Other IQ interfaces	500,000 microseconds

If you configure a delay buffer larger than the new maximum, the candidate configuration can be committed successfully. However, the setting is rejected by the packet forwarding component and a system log warning message is generated.

For interfaces that support DLCI queuing, the large buffer is supported for DLCIs on which the configured shaping rate is less than or equal to the physical interface bandwidth. For instance, when you configure a Frame Relay DLCI on a Channelized T3 IQ PIC, and you configure the shaping rate to be 1.5 Mbps, the amount of delay buffer that can be allocated to the DLCI is 500,000 microseconds, which is equivalent to a T1 delay buffer. For more information about DLCI queuing, see "Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs" on page 300.

For *N*xDS0 interfaces, the larger buffer sizes can be up to 4,000,000 microseconds, depending on the number of DS0 channels in the *N*xDS0 interface. For slower *N*xDS0 interfaces with fewer channels, the delay buffer can be relatively larger than for faster *N*xDS0 interfaces with more channels. This is shown in Table 51 on page 372.

You can allocate the delay buffer as either a percentage or a temporal value. The resulting delay buffer is calculated differently depending how you configure the delay buffer, as shown in Table 50 on page 369.

Table 50: Delay-Buffer Calculations

Delay Buffer Configuration	Formula	Example

Percentage	**available interface bandwidth * configured percentage buffer-size * maximum buffer = queue buffer**	If you configure a queue on a T1 interface to use 30 percent of the available delay buffer, the queue receives 28,125 bytes of delay buffer:
		`sched-expedited {`
		`  transmit-rate percent 30;`
		`  buffer-size percent 30;`
		`}`
		1.5 Mbps * 0.3 * 500,000 microseconds = 225,000 bits = 28,125 bytes

Temporal	**available interface bandwidth * configured percentage transmit-rate * configured temporal buffer-size = queue buffer**	If you configure a queue on a T1 interface to use 500,000 microseconds of delay buffer and you configure the transmission rate to be 20 percent, the queue receives 18,750 bytes of delay buffer:
		`sched-best {`
		`  transmit-rate percent 20;`
		`  buffer-size temporal 500000;`
		`}`
		1.5 Mbps * 0.2 * 500,000 microseconds = 150,000 bits = 18,750 bytes

Copyright © 2019, Juniper Networks, Inc.

Table 50: Delay-Buffer Calculations (continued)

Delay Buffer Configuration	Formula	Example

Percentage, with buffer size larger than transmit rate		In this example, the delay buffer is allocated twice the transmit rate. Maximum delay buffer latency can be up to twice the 500,000-microsecond delay buffer if the queue's transmit rate cannot exceed the allocated transmit rate.
		
		sched-extra-buffer {
		transmit-rate percent 10;
		buffer-size percent 20;
		}

FRF.16 LSQ bundles	For total bundle bandwidth < T1 bandwidth, the delay-buffer rate is 1 second. For total bundle bandwidth >= T1 bandwidth, the delay-buffer rate is 200 milliseconds (ms).	

Copyright © 2019, Juniper Networks, Inc.

Configuring the Maximum Delay Buffer for NxDS0 Interfaces

Copyright © 2019, Juniper Networks, Inc.

Because *N*xDS0 interfaces carry less bandwidth than a T1 or E1 interface, the buffer size on an *N*xDS0 interface can be relatively larger, depending on the number of DS0 channels combined. The maximum delay buffer size is calculated with the following formula:

```
Interface Speed * Maximum Delay Buffer Time = Delay Buffer Size
```

For example, a 1xDS0 interface has a speed of 64 kilobits per second (Kbps). At this rate, the maximum delay buffer time is 4,000,000 microseconds. Therefore, the delay buffer size is 32 kilobytes (KB):

```
64 Kbps * 4,000,000 microseconds = 32 KB
```

Table 51 on page 372 shows the delay-buffer calculations for 1xDS0 through 32xDS0 interfaces.

Table 51: NxDS0 Transmission Rates and Delay Buffers

Interface Speed	Delay Buffer Size
1xDS0 Through 4xDS0: Maximum Delay Buffer Time Is 4,000,000 Microseconds	
1xDS0: 64 Kbps	32 KB
2xDS0: 128 Kbps	64 KB
3xDS0: 192 Kbps	96 KB
4xDS0 Through 7xDS0: Maximum Delay Buffer Time Is 2,000,000 Microseconds	
4xDS0: 256 Kbps	64 KB
5xDS0: 320 Kbps	80 KB
6xDS0: 384 Kbps	96 KB
7xDS0: 448 Kbps	112 KB
8xDS0 Through 15xDS0: Maximum Delay Buffer Time Is 1,000,000 Microseconds	
8xDS0: 512 Kbps	64 KB
9xDS0: 576 Kbps	72 KB
10xDS0: 640 Kbps	80 KB
11xDS0: 704 Kbps	88 KB
12xDS0: 768 Kbps	96 KB
13xDS0: 832 Kbps	104 KB
14xDS0: 896vKbps	112 KB

Copyright © 2019, Juniper Networks, Inc.

Table 51: NxDS0 Transmission Rates and Delay Buffers (continued)

Interface Speed	Delay Buffer Size
15xDS0: 960 Kbps	120 KB
16xDS0 Through 32xDS0: Maximum Delay Buffer Time Is 500,000 Microseconds	
16xDS0: 1024 Kbps	64 KB
17xDS0: 1088 Kbps	68 KB
18xDS0: 1152 Kbps	72 KB
19xDS0: 1216 Kbps	76 KB
20xDS0: 1280 Kbps	80 KB
21xDS0: 1344 Kbps	84 KB
22xDS0: 1408 Kbps	88 KB
23xDS0: 1472 Kbps	92 KB
24xDS0: 1536 Kbps	96 KB
25xDS0: 1600 Kbps	100 KB
26xDS0: 1664 Kbps	104 KB
27xDS0: 1728 Kbps	108 KB
28xDS0: 1792 Kbps	112 KB
29xDS0: 1856 Kbps	116 KB
30xDS0: 1920 Kbps	120 KB
31xDS0: 1984 Kbps	124 KB

Copyright © 2019, Juniper Networks, Inc.

Table 51: NxDS0 Transmission Rates and Delay Buffers (continued)

Interface Speed	Delay Buffer Size
32xDS0: 2048 Kbps	128 KB

Example: Configuring Large Delay Buffers for Slower Interfaces

Set large delay buffers on interfaces configured on a Channelized OC12 IQ PIC. The CoS configuration binds a scheduler map to the interface specified in the chassis configuration. For information about the delay-buffer calculations in this example, see Table 50 on page 369.

To configure a large delay buffer:

1. Specify the FPC and PIC for which you want to configure large delay buffers.

```
[edit]
user@host# edit chassis fpc 0 pic 0
```

2. Enable large delay buffering.

```
[edit chassis fpc 0 pic 0]
user@host# set q-pic-large-buffer
```

3. Specify the maximum number of queues per interface.

```
[edit chassis fpc 0 pic 0]
user@host#  set max-queues-per-interface 8
```

4. Verify the configuration.

```
[edit chassis fpc 0 pic 0]
user@host# show
```

```
q-pic-large-buffer {
    large-scale;
}
max-queues-per-interface 8;
```

5. Save the configuration.

```
[edit chassis]
user@host# commit
```


Copyright © 2019, Juniper Networks, Inc.

Example: Configuring the Delay Buffer Value for a Scheduler

You can assign to a physical or logical interface, a scheduler map that is composed of different schedulers (or queues). The physical interface's large delay buffer can be distributed to the different schedulers (or queues) using the **transmit-rate** and **buffer-size** statements at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level.

This example shows two schedulers, **sched-best** and **sched-exped**, with the delay buffer size configured as a percentage (20 percent) and temporal value (300,000 microseconds), respectively. The **sched-best** scheduler has a transmit rate of 10 percent. The **sched-exped** scheduler has a transmit rate of 20 percent.

The **sched-best** scheduler's delay buffer is twice that of the specified transmit rate of 10 percent. Assuming that the **sched-best** scheduler is assigned to a T1 interface, this scheduler receives 20 percent of the total 500,000 microseconds of the T1 interface's delay buffer. Therefore, the scheduler receives 18,750 bytes of delay buffer:

```
available interface bandwidth * configured percentage buffer-size * maximum buffer
   = queue buffer

1.5 Mbps * 0.2 * 500,000 microseconds = 150,000 bits = 18,750 bytes
```

Assuming that the **sched-exped** scheduler is assigned to a T1 interface, this scheduler receives 300,000 microseconds of the T1 interface's 500,000-microsecond delay buffer with the traffic rate at 20 percent. Therefore, the scheduler receives 11,250 bytes of delay buffer:

```
available interface bandwidth * configured percentage transmit-rate
   * configured temporal buffer-size = queue buffer

1.5 Mbps * 0.2 * 300,000 microseconds = 90,000 bits = 11,250 bytes
```

To configure this example:

1. Configure the **sched-best** scheduler.

   ```
   [edit]
   user@host#  edit class-of-service schedulers sched-best
   ```

2. Specify the transmit-rate of 10 percent.

   ```
   [edit class-of-service schedulers sched-best]
   user@host# set transmit-rate percent 10
   ```

3. Specify the buffer size as 20 percent.

   ```
   [edit class-of-service schedulers sched-best]
   user@host#  set buffer-size percent 20
   ```

Copyright © 2019, Juniper Networks, Inc.

4. Configure the **sched-exped** scheduler.

```
[edit]
user@host# up
[edit class-of-service schedulers]
user@host# edit sched-exped
```

5. Specify the transmit-rate of 20 percent.

```
[edit class-of-service schedulers sched-exped]
user@host# set transmit-rate percent 20
```

6. Specify the buffer size temporal value (300,000 microseconds).

```
[edit class-of-service schedulers sched-exped]
user@host#  set buffer-size temporal 300000
```

7. Verify the configuration.

```
[edit]
user@host# show class-of-service
```

```
schedulers {
    sched-best {
        transmit-rate percent 10;
        buffer-size percent 20;
    }
    sched-exped {
        transmit-rate percent 20;
        buffer-size temporal 300k;
    }
}
```

8. Save the configuration.

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

Example: Configuring the Physical Interface Shaping Rate

In general, the physical interface speed is the basis for calculating the delay buffer size. However, when you include the **shaping-rate** statement, the shaping rate becomes the basis for calculating the delay buffer size. For more information, see Table 51 on page 372.

This example configures the shaping rate on a T1 interface to 200 Kbps, which means that the T1 interface bandwidth is set to 200 Kbps instead of 1.5 Mbps. Because 200 Kbps is less than 4xDS0, this interface receives 4 seconds of delay buffer, or 800 Kbps of traffic, which is 800 KB for a full second.

1. Specify the interface on which you want to configure the shaping rate..

Copyright © 2019, Juniper Networks, Inc.

```
[edit]
user@host#  edit class-of-service interfaces t1-0/0/0:1:1
```

2. Specify the shaping rate.

```
[edit class-of-service interfaces t1-0/0/0:1:1]
user@host# set shaping-rate 200k
```

3. Verify the configuration.

```
[edit class-of-service]
user@host# show
```

```
interfaces {
    t1-0/0/0:1:1 {
        shaping-rate 200k;
    }
}
```

4. Save the configuration.

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

Complete Configuration

This example shows a Channelized OC12 IQ PIC in FPC slot 0, PIC slot 0 and a channelized T1 interface with Frame Relay encapsulation. It also shows a scheduler map configuration on the physical interface.

```
chassis {
  fpc 0 {
    pic 0 {
      q-pic-large-buffer;
      max-queues-per-interface 8;
    }
  }
}
interfaces {
  coc12-0/0/0 {
    partition 1 oc-slice 1 interface-type coc1;
  }
  coc1-0/0/0:1 {
    partition 1 interface-type t1;
  }
  t1-0/0/0:1:1 {
    encapsulation frame-relay;
    unit 0 {
      family inet {
```

Copyright © 2019, Juniper Networks, Inc.

```
            address 10.1.1.1/24;
          }
          dlci 100;
        }
      }
    }
    class-of-service {
      interfaces {
        t1-0/0/0:1:1 {
          scheduler-map smap-1;
        }
      }
      scheduler-maps {
        smap-1 {
          forwarding-class best-effort scheduler sched-best;
          forwarding-class expedited-forwarding scheduler sched-exped;
          forwarding-class assured-forwarding scheduler sched-assure;
          forwarding-class network-control scheduler sched-network;
        }
      }
      schedulers {
        sched-best {
          transmit-rate percent 40;
          buffer-size percent 40;
        }
        sched-exped {
          transmit-rate percent 30;
          buffer-size percent 30;
        }
        sched-assure {
          transmit-rate percent 20;
          buffer-size percent 20;
        }
        sched-network {
          transmit-rate percent 10;
          buffer-size percent 10;
        }
      }
    }
```

Enabling and Disabling the Memory Allocation Dynamic per Queue

In the Junos OS, the memory allocation dynamic (MAD) is a mechanism that dynamically provisions extra delay buffer when a queue is using more bandwidth than it is allocated in the transmit rate setting. With this extra buffer, queues absorb traffic bursts more easily, thus avoiding packet drops. The MAD mechanism can provision extra delay buffer only when extra transmission bandwidth is being used by a queue. This means that the queue might have packet drops if there is no surplus transmission bandwidth available.

For Juniper Networks M320 Multiservice Edge Routers, MX Series 5G Universal Routing Platforms, T Series Core Routers, and EX Series Ethernet Switches only, the MAD mechanism is enabled unless the delay buffer is configured with a temporal setting for a given queue. The MAD mechanism is particularly useful for forwarding classes carrying latency-immune traffic for which the primary requirement is maximum bandwidth

Copyright © 2019, Juniper Networks, Inc.

utilization. In contrast, for latency-sensitive traffic, you might wish to disable the MAD mechanism because large delay buffers are not optimum.

MAD support is dependent on the FPC and Packet Forwarding Engine, not the PIC. All M320, MX Series, and T Series router and EX Series switches' FPCs and Packet Forwarding Engines support MAD. No Modular Port Concentrators (MPCs) and IQ, IQ2, IQ2E or IQE PICs support MAD.

To enable the MAD mechanism on supported hardware:

1. Include the **buffer-size percent** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level:

```
[edit class-of-service schedulers scheduler-name]
user@host# set buffer-size percent percentage
```

The minimum buffer allocated to any queue is 18,432 bytes. If a queue is configured to have a buffer size less than 18K, the queue retains a buffer size of 18,432 bytes.

If desired, you can configure a buffer size that is greater than the configured transmission rate. The buffer can accommodate packet bursts that exceed the configured transmission rate, if sufficient excess bandwidth is available. For example:

```
class-of-service {
  schedulers {
    sched-best {
      transmit-rate percent 20;
      buffer-size percent 30;
    }
  }
}
```

As stated previously, you can use a temporal delay buffer configuration to disable the MAD mechanism on a queue, thus limiting the size of the delay buffer. However, the effective buffer latency for a temporal queue is bounded not only by the buffer size value but also by the associated drop profile. If a drop profile specifies a drop probability of 100 percent at a fill-level less than 100 percent, the effective maximum buffer latency is smaller than the buffer size setting. This is because the drop profile specifies that the queue drop packets before the queue's delay buffer is 100 percent full.

Such a configuration might look like the following example:

```
class-of-service {
  drop-profiles {
    plp-high {
      fill-level 70 drop-probability 100;
    }
    plp-low {
      fill-level 80 drop-probability 100;
    }
  }
```

Copyright © 2019, Juniper Networks, Inc.

```
  schedulers {
    sched {
      buffer-size temporal 500000;
      drop-profile-map loss-priority low protocol any drop-profile plp-low;
      drop-profile-map loss-priority high protocol any drop-profile plp-high;
      transmit-rate percent 20;
    }
  }
}
```

Related Documentation

- buffer-size (Schedulers) on page 1056
- schedulers (CoS) on page 1262
- *q-pic-large-buffer*
- schedulers (CoS) on page 1262

Copyright © 2019, Juniper Networks, Inc.

Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy

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

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

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

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

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

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

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

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

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

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

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

To configure weighted average buffer occupancy:

Copyright © 2019, Juniper Networks, Inc.

1. Specify the FPC slot number and Q-PIC number on which you want to configure RED weighted average buffer occupancy calculations:

```
[edit]
user@host# edit chassis fpc slot-number pic pic-number red-buffer-occupancy
  weighted-averaged
```

2. Specify the weight exponent value.

```
[edit chassis fpc slot-number pic pic-number red-buffer-occupancy weighted-averaged]
user@host# set instant-usage-weight-exponent exponent-value
```

3. Verify your configuration.

```
[edit chassis]
user@host# show
```

For example:

```
[edit chassis]
user@host# show
fpc 1 {
    pic 1 {
        red-buffer-occupancy {
            weighted-averaged {
                instant-usage-weight-exponent 3;
            }
        }
    }
}
```

4. Save your configuration.

```
[edit chassis]
user@host# commit
```

Related Documentation

- Example: Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 383
- red-buffer-occupancy on page 1244

Copyright © 2019, Juniper Networks, Inc.

Example: Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy

This topic provides two examples for configuring the weighted RED buffer occupancy feature to manage traffic bursts.

- Requirements on page 383
- Overview on page 383
- Configuration on page 383

Requirements

Weighted RED buffer occupancy is configured on a per-PIC basis and applies to only the following IQ-PICs:

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

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

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

Overview

To manage traffic bursts on IQ PICs, you can base RED queue management on weighted average buffer occupancy values. This topic provides two examples for configuring weighted RED buffer occupancy feature to manage traffic bursts.

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, and then copy and paste the commands into the CLI.

To configure a Q-PIC to use a weight value of 1/2 in average buffer occupancy calculations:

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

To configure a Q-PIC to use a weight value of 1/4 in average buffer occupancy calculations:

```
[edit]
edit chassis fpc 0 pic 1
```

Copyright © 2019, Juniper Networks, Inc.

```
set red-buffer-occupancy weighted-averaged instant-usage-weight-exponent 2
```

Example: Configuring a Q-PIC to Use a Weight Value of 1/2 in Average Buffer Occupancy Calculations

Step-by-Step Procedure

To configure a Q-PIC to use a weight value of 1/2 in average buffer occupancy calculations:

1. Specify the Q-PIC.

```
[edit]
user@host# edit chassis fpc 1 pic 0
```

2. Configure the RED queue management values.

```
[edit chassis fpc 1 pic 0]
user@host# set red-buffer-occupancy weighted-averaged
  instant-usage-weight-exponent 1
```

Example: Configuring a Q-PIC to Use a Weight Value of 1/4 in Average Buffer Occupancy Calculations

Step-by-Step Procedure

To configure a Q-PIC to use a weight value of 1/4 in average buffer occupancy calculations:

1. Specify the Q-PIC.

```
[edit]
user@host# edit chassis fpc 1 pic 1
```

2. Configure the RED queue management values.

```
[edit chassis fpc 1 pic 1]
user@host# set red-buffer-occupancy weighted-averaged
  instant-usage-weight-exponent 2
```

Results

From configuration mode, confirm your configuration by entering the **show** command at the **[edit chassis fpc 1]** hierarchy level. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit chassis fpc 1]
user@host# show
```

```
red-buffer-occupancy {
    weighted-averaged {
```


Copyright © 2019, Juniper Networks, Inc.

```
        instant-usage-weight-exponent 1;
    }
}
red-buffer-occupancy {
    weighted-averaged {
        instant-usage-weight-exponent 2;
    }
}
```

Enter **commit** from configuration mode.

Related Documentation

- Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 381
- red-buffer-occupancy on page 1244

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 10

Altering Outgoing Packet Headers Using Rewrite Rules

- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388
- Applying Default Rewrite Rules on page 389
- Configuring Rewrite Rules on page 391
- Configuring Rewrite Rules Based on PLP on page 392
- Applying IEEE 802.1p Rewrite Rules to Dual VLAN Tags on page 393
- Applying IEEE 802.1ad Rewrite Rules to Dual VLAN Tags on page 395
- Rewriting IEEE 802.1p Packet Headers with an MPLS EXP Value on page 396
- Setting IPv6 DSCP and MPLS EXP Values Independently on page 398
- Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 398
- Setting Ingress DSCP Bits for Multicast Traffic over Layer 3 VPNs on page 400
- Applying Rewrite Rules to Output Logical Interfaces on page 401
- Rewriting MPLS and IPv4 Packet Headers on page 403
- Rewriting the EXP Bits of All Three Labels of an Outgoing Packet on page 407
- Defining a Custom Frame Relay Loss Priority Map on page 409
- Example: Per-Node Rewriting of EXP Bits on page 410
- Example: Rewriting CoS Information at the Network Border to Enforce CoS Strategies on page 411
- Example: Remarking Diffserv Code Points to MPLS EXPs to Carry CoS Profiles Across a Service Provider's L3VPN MPLS Network on page 422
- Example: Remarking Diffserv Code Points to 802.1P PCPs to Carry CoS Profiles Across a Service Provider's VPLS Network on page 446
- Assigning Rewrite Rules on a Per-Customer Basis Using Policy Maps Overview on page 469
- Configuring Policy Maps to Assign Rewrite Rules on a Per-Customer Basis on page 471

Copyright © 2019, Juniper Networks, Inc.

Rewriting Packet Headers to Ensure Forwarding Behavior

As packets enter or exit a network, edge routers might be required to alter the class-of-service (CoS) settings of the packets. Rewrite rules set the value of the CoS bits within the packet's header. Each rewrite rule reads the current forwarding class and loss priority information associated with the packet, locates the chosen CoS value from a table, and writes this CoS value into the packet header.

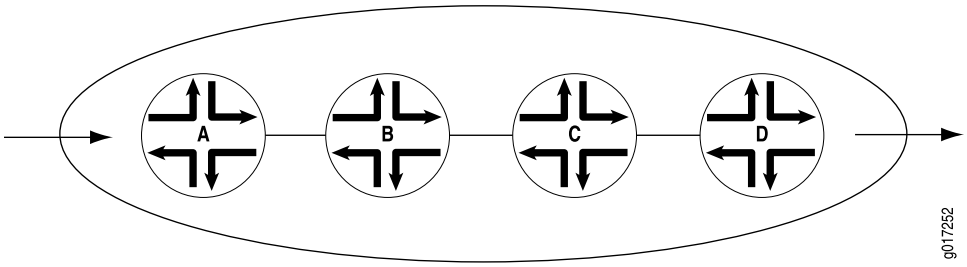
In effect, the rewrite rule performs the opposite function of the behavior aggregate (BA) classifier used when the packet enters the routing device. As the packet leaves the routing platform, the final CoS action is generally the application of a rewrite rule.

You configure rewrite rules to alter CoS values in outgoing packets on the outbound interfaces of an edge router to meet the policies of a targeted peer. This allows the downstream routing device in a neighboring network to classify each packet into the appropriate service group.

In addition, you often need to rewrite a given marker (IP precedence, Differentiated Services code point [DSCP], IEEE 802.1p, or MPLS EXP settings) at the inbound interfaces of an edge router to accommodate BA classification by core devices.

Figure 40 on page 388 shows a flow of packets through four routing devices. Router A rewrites the CoS bits in incoming packet to accommodate the BA classification performed by Routers B and C. Router D alters the CoS bits of the packets before transmitting them to the neighboring network.

Figure 40: Packet Flow Across the Network



For every incoming packet, the ingress classifier decodes the ingress CoS bits into a forwarding class and packet loss priority (PLP) combination. The egress CoS information depends on which type of rewrite marker is active, as follows:

- For Multiprotocol Label Switching (MPLS) EXP and IEEE 802.1 rewrite markers, values are derived from the forwarding class and PLP values in rewrite rules. MPLS EXP and IEEE 802.1 markers are not preserved because they are part of the Layer 2 encapsulation.
- For IP precedence and DiffServ code point (DSCP) rewrite markers, the marker alters the first three bits on the type-of-service (ToS) byte while leaving the last three bits unchanged.

Copyright © 2019, Juniper Networks, Inc.

To configure CoS rewrite rules, you define the rewrite rule and apply it to an interface. Include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
interfaces {
  interface-name {
    unit logical-unit-number {
      rewrite-rules {
        dscp (rewrite-name | default)protocol protocol-types;
        dscp-ipv6 (rewrite-name | default);
        exp (rewrite-name | default)protocol protocol-types;
        exp-push-push-push default;
        exp-swap-push-push default;
        ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
        ieee-802.1ad (rewrite-name | default) vlan-tag (outer | outer-and-inner);
        inet-precedence (rewrite-name | default)protocol protocol-types;
      }
    }
  }
}
rewrite-rules {
  (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | inet-precedence) rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
      loss-priority level code-point (alias | bits);
    }
  }
}
```

Applying Default Rewrite Rules

By default, rewrite rules are not usually applied to interfaces. If you want to apply a rewrite rule, you can either design your own rule and apply it to an interface, or you can apply a default rewrite rule.

NOTE: The lone exception is that non-MPC MPLS-enabled interfaces use the default EXP rewrite rule, even if not configured.

To apply default rewrite rules, include one or more of the following statements at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]
dscp default;
dscp-ipv6 default;
exp default;
ieee-802.1 default vlan-tag (outer | outer-and-inner);
inet-precedence default;
```

Table 52 on page 390 shows the default rewrite rule mappings. These are based on the default bit definitions of DSCP, DSCP IPv6, EXP, IEEE, and IP CoS values, as shown in

Copyright © 2019, Juniper Networks, Inc.

"Default Aliases for CoS Value Bit Patterns Overview" on page 45, and the default forwarding classes shown in "Default Forwarding Classes" on page 208.

When the software detects packets whose CoS values match the forwarding class and PLP values in the first two columns in Table 52 on page 390, the software maps the header bits of those packets to the code-point aliases in the last column in Table 52 on page 390. The code-point aliases in the last column map to the CoS bits shown in "Default Aliases for CoS Value Bit Patterns Overview" on page 45.

Table 52: Default Packet Header Rewrite Mappings

Map from Forwarding Class	PLP Value	Map to DSCP/DSCP IPv6/ EXP/IEEE/IP
expedited-forwarding	**low**	**ef**
expedited-forwarding	**high**	**ef**
assured-forwarding	**low**	**af11**
assured-forwarding	**high**	**af12** (DSCP/DSCP IPv6/EXP)
best-effort	**low**	**be**
best-effort	**high**	**be**
network-control	**low**	**nc1/cs6**
network-control	**high**	**nc2/cs7**

In the following example, the **so-1/2/3.0** interface is assigned the default DSCP rewrite rule. One result of this configuration is that each packet exiting the interface with the **expedited-forwarding** forwarding class and the **high** or **low** loss priority has its DSCP bits rewritten to the DSCP **ef** code-point alias. "Default Aliases for CoS Value Bit Patterns Overview" on page 45 shows that this code-point alias maps to the **101110** bits.

Another result of this configuration is that all packets exiting the interface with the **best-effort** forwarding class and the **high** or **low** loss priority have their EXP bits rewritten to the EXP **be** code-point alias."Default Aliases for CoS Value Bit Patterns Overview" on page 45 shows that this code-point alias maps to the **000** bits.

To evaluate all the implications of this example, see "Default Aliases for CoS Value Bit Patterns Overview" on page 45 and Table 52 on page 390.

```
class-of-service {
  interfaces {
    so-1/2/3 {
      unit 0 {
        rewrite-rules {
          dscp default;
        }
```


Copyright © 2019, Juniper Networks, Inc.

```
        }
      }
    }
  }
```

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36

Configuring Rewrite Rules

You define markers in the rewrite rules section of the CoS configuration hierarchy and reference them in the logical interface configuration. This model supports marking on the DSCP, DSCP IPv6, IP precedence, IPv6 precedence, IEEE 802.1, and MPLS EXP CoS values.

To configure a rewrite-rules mapping and associate it with the appropriate forwarding class and code-point alias or bit set, include the **rewrite-rules** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
rewrite-rules {
  (dscp | dscp-ipv6 | exp | ieee-802.1 |ieee-802.1ad | inet-precedence | inet6-precedence)
    rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
      loss-priority level code-point (alias | bits);
    }
  }
}
```

NOTE: The inet-precedence statement is supported on PTX Series routers only when network services is set to enhanced-mode. For more information, see enhanced-mode.

The rewrite rule sets the code-point aliases and bit patterns for a specific forwarding class and packet loss priority (PLP). The inputs for the map are the forwarding class and the PLP. The output of the map is the code-point alias or bit pattern. For more information about how CoS maps work, see "Mapping CoS Component Inputs to Outputs" on page 11.

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

Copyright © 2019, Juniper Networks, Inc.

Starting with Junos OS Release 18.1R1, MX Series routers with MPCs support rewrite rules that rewrite the first three bits of the IPv6 DSCP value through the **inet6-precedence** statement at the **[edit class-of-service rewrite-rules]** hierarchy level. This allows you to set a 3-bit code point for a particular forwarding class and loss priority for IPv6 traffic. This rewrite rule option can also be applied to packets entering an MPLS LSP by including the protocol **mpls statement** when applying the rewrite rule to a logical interface.

On the M320, T1600, and MX960 routers and EX Series switches, if you configure **vlan-vpls** encapsulation and add an IEEE 802.1 header on a Gigabit Ethernet or 10 Gigabit Ethernet interface to output traffic, but do not apply an IEEE 802.1 rewrite rule, then the default IEEE 802.1 rewrite rule is ignored and the IEEE 802.1p bits are set to match the forwarding class queue.

Integrated Bridging and Routing (IRB) interfaces are used to tie together Layer 2 switched and Layer 3 routed domains on MX routers. MX routers support classifiers and rewrite rules on the IRB interface at the **[edit class-of-service interfaces irb unit logical-unit-number]** level of the hierarchy. All types of classifiers and rewrite rules are allowed, including IEEE 802.1p.

NOTE: The IRB classifiers and rewrite rules are used only for *routed* packets; in other words, it is for traffic that originated in the Layer 2 domain and is then routed through IRB into the Layer 3 domain, or vice versa. Only IEEE classifiers and IEEE rewrite rules are allowed for pure Layer 2 interfaces within a bridge domain.

NOTE: The forwarding class and loss priority are determined by ingress classification.

Release History Table

Release	Description
18.1	Starting with Junos OS Release 18.1R1, MX Series routers with MPCs support rewrite rules that rewrite the first three bits of the IPv6 DSCP value through the **inet6-precedence** statement at the **[edit class-of-service rewrite-rules]** hierarchy level.

Related Documentation

- Applying Rewrite Rules to Output Logical Interfaces on page 401
- Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 597

Configuring Rewrite Rules Based on PLP

Rewrite rules take action on outgoing packets. When tricolor marking (TCM) is enabled, routers support four rewrite packet loss priority (PLP) designations: **low**, **medium-low**,

Copyright © 2019, Juniper Networks, Inc.

medium-high, and **high**. To include the PLP for a rewrite rule, include the following statements at the **[edit class-of-service]** hierarchy level:

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

In Junos OS, rewrite rules only look at the forwarding class and packet loss priority of the packet (as assigned by a behavior aggregate or multifield classifier at ingress), not at the incoming CoS value, to determine the CoS value to write to the packet header at egress. The inputs for a rewrite rule are the forwarding class and the PLP. The output for a rewrite rule are the CoS values. In other words, a rewrite rule sets the CoS value for each packet exiting the interface with a specified forwarding class and PLP.

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

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

To use this classifier, you must configure the settings for the **assured-forwarding** forwarding class at the **[edit class-of-service forwarding-classes queue *queue-number* assured-forwarding]** hierarchy level. For more information, see "Understanding How Forwarding Classes Assign Classes to Output Queues" on page 205.

Applying IEEE 802.1p Rewrite Rules to Dual VLAN Tags

By default, when you apply an IEEE 802.1p rewrite rule to an output logical interface, the software rewrites the IEEE bits in the outer VLAN tag only.

For Gigabit Ethernet IQ2 PICs, 10-port 10-Gigabit OSE PICs, and 10-Gigabit Ethernet IQ2 PICs only, you can rewrite the IEEE bits in both the outer and inner VLAN tags of the tagged Ethernet frames. When you enable class of service (CoS) rewrite for both tags, the same IEEE 802.1p rewrite table is used for the inner and outer VLAN tag.

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

Copyright © 2019, Juniper Networks, Inc.

To rewrite both the outer and inner VLAN tags, include the **vlan-tag outer-and-inner** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules ieee-802.1 (*rewrite-name* | default)]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules
  ieee-802.1 (rewrite-name | default)]
vlan-tag outer-and-inner;
```

To explicitly specify the default behavior, include the **vlan-tag outer** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules ieee-802.1 (*rewrite-name* | default)]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules
  ieee-802.1 (rewrite-name | default)]
vlan-tag outer;
```

For more information about VLAN tags, see the *Junos OS Network Interfaces Library for Routing Devices*.

On MX routers and EX Series switches, you can perform IEEE 802.1p and DEI rewriting based on forwarding class and PLP at the VPLS ingress PE. You rewrite (mark) the IEEE 802.1p or DEI bits on frames at the VPLS ingress PE based on the value of the forwarding class and PLP established for the traffic. You can rewrite either the outer tag only or the outer and inner tag. When both tags are rewritten, both get the same value. To configure these rewrite rules, include the **ieee-802.1** statement at the [**edit class-of-services routing-instance *routing-instance-name* rewrite-rules**] hierarchy level.

NOTE: For MX80, MX240, MX480, and MX960 routers with MPC/MICs, rewrite on LSI interfaces is not supported (the routers, with DPC, do support rewrite on LSI interfaces).

On routing devices with IQ2 or IQ2-E PICs, you can perform IEEE 802.1p and DEI rewriting based on forwarding-class and packet loss priority (PLP) at the VPLS ingress provider edge (PE) router. You rewrite (mark) the IEEE 802.1p or DEI bits on frames at the VPLS ingress PE based on the value of the forwarding-class and PLP established for the traffic. You can rewrite either the outer tag only or both the outer and inner tags. When both tags are rewritten, both get the same value.

NOTE: The 10-port 10-Gigabit OSE PIC does not support DEI rewriting based on forwarding class and PLP at the VPLS ingress PE.

To configure these rewrite rules, include the **ieee-802.1** statement at the [**edit class-of-services routing-instance *routing-instance-name* rewrite-rules**] hierarchy level.

Example: Applying an IEEE 802.1p Rewrite Rule to Dual VLAN Tags

Apply the **ieee8021p-rwrule1** rewrite rule to both inner and outer VLAN tags of Ethernet-tagged frames exiting the **ge-0/0/0.0** interface:

Copyright © 2019, Juniper Networks, Inc.

```
class-of-service {
  interfaces {
    ge-0/0/0 {
      unit 0 {
        rewrite-rules {
          ieee-802.1 ieee8021p-rwrule1 vlan-tag outer-and-inner;
        }
      }
    }
  }
}
```

Applying IEEE 802.1ad Rewrite Rules to Dual VLAN Tags

By default, when you apply an IEEE 802.1ad rewrite rule to an output logical interface, the software rewrites the IEEE bits in the outer VLAN tag only.

For MX Series routers and IQ2 PICs, you can rewrite the IEEE 802.1ad bits in both the outer and inner VLAN tags of the tagged Ethernet frames. When you enable the CoS rewrite for both tags, the same IEEE 802.1ad rewrite table is used for the inner and outer VLAN tag.

NOTE: When you apply IEEE 802.1ad rewrite rules for inner and outer VLAN tags, you cannot rewrite the Canonical Format Identifier (CFI) bit for the inner VLAN tag.

To rewrite both the outer and inner VLAN tags, include the **vlan-tag outer-and-inner** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules ieee-802.1ad (*rewrite-name* | default)]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules
  ieee-802.1ad (rewrite-name | default)]
vlan-tag outer-and-inner;
```

To explicitly specify the default behavior, include the **vlan-tag outer** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules ieee-802.1ad (*rewrite-name* | default)]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules
  ieee-802.1ad (rewrite-name | default)]
vlan-tag outer;
```

For more information about VLAN tags, see the *Junos OS Network Interfaces Library for Routing Devices*.

Example: Applying an IEEE 802.1ad Rewrite Rule to Dual VLAN Tags

Apply the **dot1p_dei_rw** rewrite rule to both inner and outer VLAN tags of Ethernet-tagged frames exiting the **ge-2/0/0.0** interface:

Copyright © 2019, Juniper Networks, Inc.

```
class-of-service {
  interfaces {
    ge-2/0/0 {
      unit 0 {
        rewrite-rules {
          ieee-802.1ad dot1p_dei_rw vlan-tag outer-and-inner;
        }
      }
    }
  }
}
```

Rewriting IEEE 802.1p Packet Headers with an MPLS EXP Value

For Ethernet interfaces on Juniper Networks M320 Multiservice Edge Routers, MX Series Ethernet Service Routers, T Series Core Routers, and EX Series switches that have a peer connection to an M Series Multiservice Edge Router, MX Series router, T Series router, or EX Series switch, you can rewrite both MPLS EXP and IEEE 802.1p bits to a configured value. This enables you to pass the configured value to the Layer 2 VLAN path. For IQ PICs, you can only configure one IEEE 802.1 rewrite rule on a physical port. All logical ports (units) on that physical port should apply the same IEEE 802.1 rewrite rule.

To rewrite both the MPLS EXP and IEEE 802.1p bits, you must include EXP and IEEE 802.1p rewrite rules in the interface configuration. To configure EXP and IEEE 802.1p rewrite rules, include the **rewrite-rules** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level, specifying the **exp** and **ieee-802.1** options:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
rewrite-rules {
  exp rewrite-rule-name;
  ieee-802.1 default;
}
```

When you combine these two rewrite rules, only the EXP rewrite table is used for rewriting packet headers. If you do not configure a VLAN on the interface, only the EXP rewriting is in effect. If you do not configure an LSP on the interface or if the MPLS EXP rewrite rule mapping is removed, the IEEE 802.1p default rewrite rules mapping takes effect.

Copyright © 2019, Juniper Networks, Inc.

NOTE: You can also combine other rewrite rules. IP, DSCP, DSCP IPv6, and MPLS EXP are associated with Layer 3 packet headers, and IEEE 802.1p is associated with Layer 2 packet headers.

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

If you combine IEEE 802.1p with IP rewrite rules, the Layer 3 packets and Layer 2 headers are rewritten with the IP rewrite rule.

If you combine IEEE 802.1p with DSCP or DSCP IPv6 rewrite rules, three bits of the Layer 2 header and six bits of the Layer 3 packet header are rewritten with the DSCP or DSCP IPv6 rewrite rule.

NOTE: For MPCs, default EXP rewrite rules do not exist for logical interfaces. The EXP CoS bits for MPLS labels are obtained from the IP precedence bits for IP traffic. The EXP bits for labels that are pushed or swapped are inherited from the current label of the MPLS packets. For non-IP and non-MPLS packets, the EXP bits are set to 0. If a custom EXP rewrite rule is configured on the core-facing interface, then it overrides the EXP bits.

The following example shows how to configure an EXP rewrite rule and apply it to both MPLS EXP and IEEE 802.1p bits:

```
[edit class-of-service]
rewrite-rules {
   exp exp-ieee-table {
      forwarding-class best-effort {
         loss-priority low code-point 000;
         loss-priority high code-point 001;
      }
      forwarding-class assured-forwarding {
         loss-priority low code-point 010;
         loss-priority high code-point 011;
      }
      forwarding-class expedited-forwarding {
         loss-priority low code-point 111;
         loss-priority high code-point 110;
      }
      forwarding-class network-control {
         loss-priority low code-point 100;
         loss-priority high code-point 101;
      }
   }
}
interfaces {
   so-3/1/0 {
      unit 0 {
```

Copyright © 2019, Juniper Networks, Inc.

```
        rewrite-rules {
          exp exp-ieee-table;
          ieee-802.1 default;
        }
      }
    }
  }
```

Setting IPv6 DSCP and MPLS EXP Values Independently

On the M120, M320 with Enhanced III FPCs, MX Series 5G Universal Routing Platforms, and EX Series switches, you can set the DSCP and MPLS EXP bits independently on IPv6 packets. To enable this feature, include the **protocol mpls** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules dscp-ipv6 *rewrite-name*]** hierarchy level.

You can set DSCP IPv6 values only at the ingress MPLS node.

The following limitations apply to this feature:

- This feature is supported only on M120, M320 with Enhanced III FPCs, MX Series Ethernet Services routers, and EX Series switches.
- MPLS packets entering another MPLS tunnel at the ingress node may mark only the EXP value if EXP rewrite rules are configured, but not the DSCP value in the IPv6 header.
- This feature does not support MPLS packets generated by the Routing Engine.
- The IP precedence field is not applicable for IPv6, and is not supported.

Related Documentation

- Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 398

Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel

The following configuration example explains in detail how to set the DSCP and MPLS EXP bits independently on IPv6 packets.

1. Configure the router device (ingress PE router) to classify (behavior aggregate or multifield) the incoming packets to a particular forwarding class.

```
[edit firewall]
family inet6 {
  filter ss-v6filt {
    term ss-vpn {
      from {
        destination-address {
          ::ffff:192.0.2.128/120;
        }
      }
        then {
        loss-priority low;
```


Copyright © 2019, Juniper Networks, Inc.

```
            forwarding-class ss-fc;
        }
    }
  }
}
```

In the preceding example, the ingress FPC classifies (MF) incoming IPv6 packets destined for address "::ffff:192.0.2.128/120" to forwarding class "ss-fc" and loss priority "low."

2. Attach the preceding firewall filter to an interface. Because you are matching on inbound traffic, this would be an input filter. This classifies all traffic on the interface "ge-2/1/0" that matches the filter "ss-v6."

```
[edit interfaces]
ge-2/1/0 {
  hierarchical-scheduler;
  vlan-tagging;
  unit 300 {
    family inet6 {
      filter {
        input ss-v6filt;
      }
      address ::ffff:192.0.2.100/120;
    }
  }
}
```

3. Configure the DSCP–IPv6 rewrite rule for the forwarding class "ss-fc." This causes the outgoing IPv6 packets belonging to the forwarding class "ss-fc" and loss priority "low" to have their DSCP value rewritten to 100000.

```
[edit class-of-service rewrite-rules]
dscp-ipv6 ss-v6dscp {
  forwarding-class ss-fc {
    loss-priority low code-point 100000;
  }
}
```

4. Configure the EXP rewrite values for the forwarding class "ss-fc." This rewrite rule stamps an EXP value of 100 on all outgoing MPLS packets assigned to the forwarding class "ss-fc" and loss priority "low."

```
[edit class-of-service rewrite-rules]
exp ss-exp {
  forwarding-class ss-fc {
    loss-priority low code-point 100;
  }
}
```

5. Apply the preceding rewrite rule to an egress interface. On the egress FPC, all IPv6 packets in the forwarding class "ss-fc" with loss priority "low" are marked with the DSCP value "100000" and an EXP value of "100" before they enter the MPLS tunnel.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces]
ge-2/1/1 {
  unit 10 {
    rewrite-rules {
      dscp-ipv6 ss-v6dscp protocol mpls;
      exp ss-exp;
    }
  }
}
```

6. To support IPv4 DSCP and MPLS EXP independent rewrite at the same time, you can define and apply an IPv4 DSCP rewrite rule “ss-dscp” to the same interface.

```
[edit class-of-service interfaces]
ge-2/1/1 {
  unit 10 {
    rewrite-rules {
      dscp ss-dscp protocol mpls;
      dscp-ipv6 ss-v6dscp protocol mpls;
      exp ss-exp;
    }
  }
}
```

Related Documentation

- Setting IPv6 DSCP and MPLS EXP Values Independently on page 398

Setting Ingress DSCP Bits for Multicast Traffic over Layer 3 VPNs

By default, the DSCP bits on outer IP headers arriving at an ingress PE router using generic routing encapsulation (GRE) are not set for multicast traffic sent over an Layer 3 virtual private network (VPN) provider network. However, you can configure a type-of-service (ToS) rewrite rule so the router sets the DSCP bits of GRE packets to be consistent with the service provider’s overall core network CoS policy. The bits are set at the core-facing interface of the ingress provider edge (PE) router. For more information about rewriting IP header bits, see “Rewriting Packet Headers to Ensure Forwarding Behavior” on page 388.

This section describes this configuration from a CoS perspective. The examples are not complete multicast or VPN configurations. For more information about multicast, see the *Multicast Protocols Feature Guide* . For more information about Layer 3 VPNs, see the *Junos OS VPNs Library for Routing Devices*.

To configure the rewrite rules on the core-facing interface of the ingress PE, include the **rewrite-rules** statement at the **[edit class-of-service]** hierarchy level. You apply the rule to the proper ingress interface at the **[edit class-of-service interfaces]** hierarchy level to complete the configuration. This ingress DSCP rewrite is independent of classifiers placed on ingress traffic arriving on the customer-facing interface of the PE router.

The rewrite rules are applied to all unicast packets and multicast groups. You cannot configure different rewrite rules for different multicast groups. The use of DSCPv6 bits

Copyright © 2019, Juniper Networks, Inc.

is not supported because IPv6 multicast is not supported. You can configure another rewrite rule for the EXP bits on MPLS CE-CE unicast traffic.

This example defines a rewrite rule called **dscp-rule** that establishes a value of **000000** for best-effort traffic. The rule is applied to the outgoing, core-facing PE interface **ge-2/3/0**.

```
[edit class-of-service]
rewrite-rules {
   dscp dscp-rule {
      forwarding-class best-effort {
         loss-priority low code-point 000000;
      }
   }
}

[edit class-of-service interfaces]
ge-2/3/0 {
   unit 0 {
      rewrite-rules {
         dscp dscp-rule;
      }
   }
}
```

Applying Rewrite Rules to Output Logical Interfaces

To assign the rewrite-rules configuration to the output logical interface, include the **rewrite-rules** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
rewrite-rules {
   dscp (rewrite-name | <default>)protocol protocol-types;
   dscp-ipv6 (rewrite-name | <default>) protocol protocol-types
   exp (rewrite-name | <default>)protocol protocol-types;
   exp-push-push-push <default>;
   exp-swap-push-push <default>;
   ieee-802.1 (rewrite-name | <default>) inet-prec vlan-tag (outer | outer-and-inner);
   inet-precedence (rewrite-name | <default>)protocol protocol-types;
   inet6-precedence (rewrite-name | <default>)protocol protocol-types;
}
```

On M120, M320 with an Enhanced III FPC, MX Series routers and T 4000 routers with Type 5 FPCs and EX Series switches, you can combine the **dscp** or **inet-prec** and **exp** options to set the DSCP or IP precedence bits and MPLS EXP bits independently on IP packets entering an MPLS tunnel.

For IQ PICs, you can configure only one IEEE 802.1 rewrite rule on a physical port. All logical ports (units) on that physical port should apply the same IEEE 802.1 rewrite rule. If you configure more than one IEEE 802.1 rewrite rule for the IQ PIC, the configuration check fails.

Copyright © 2019, Juniper Networks, Inc.

Logical interfaces do not support multiple **dscp** rewrite rules or multiple **dscp-ipv6** rewrite rules for the same protocol.

In the following example, the DSCP bits specified in **ss-dscp** are applied to packets entering the MPLS tunnel on **ge-2/1/1**, and the DSCP bits specified in ss-v6dscp are applied to IPv6 packets. The EXP bits are set to the bit configuration specified in **ss-exp**:

```
[edit class-of-service interfaces]
ge-2/1/1
   unit 10 {
      rewrite-rules {
         dscp ssf-dscp protocol mpls; # Applies to IPv4 packets entering MPLS tunnel
         dscp-ipv6 ss-v6dscp protocol mpls; # Applies to IPv6 packets entering MPLS tunnel
         exp ss-exp; # Sets label EXP bits independently
      }
   }
}
```

You can use interface wildcards for ***interface-name*** and ***logical-unit-number***. You can also include Layer 2 and Layer 3 rewrite information in the same configuration.

Copyright © 2019, Juniper Networks, Inc.

NOTE: On M Series routers only, if you include the **control-word** statement at the **[edit protocols l2circuit neighbor** *address* **interface** *interface-name***]** hierarchy level, the software cannot rewrite MPLS EXP bits.

DSCP and DSCP IPv6 rewrite rules are supported on M Series and T Series routers when non-queuing PICs are installed, but are disabled when queuing PICs are installed with the following exceptions:

- On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.
- On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.

DSCP and DCSP IPv6 rewrite rules are supported on MX Series routers with MPC/MIC interfaces and on EX Series switches.

Inet6 precedence rewrite rules are supported on MX Series routers with MPC/MIC interfaces.

DSCP rewrite rules are not supported on T Series routers when IQ, IQ2, IQE, IQ2E, SONET/SDH OC48/STM16 IQE, or PD-5-10XGE-SFPP PICs are installed.

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

On M320 and T Series routers (except for T4000 routers with Type 5 FPCs), for a single interface, you cannot enable a rewrite rule on a subset of forwarding classes. You must assign a rewrite rule to either none of the forwarding classes or all of the forwarding classes. When you assign a rewrite rule to a subset of forwarding classes, the commit does not fail, and the subset of forwarding classes works as expected. However, the forwarding classes to which the rewrite rule is not assigned are rewritten to all zeros.

For example, if you configure a Differentiated Services code point (DSCP) rewrite rule, the bits in the forwarding classes to which you do not assign the rewrite rule are rewritten to 000000; if you configure an IP precedence rewrite rule, the bits in the forwarding classes to which you do not assign the rewrite rule are rewritten to 000.

Related Documentation

- Setting IPv6 DSCP and MPLS EXP Values Independently on page 398
- Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 398

Rewriting MPLS and IPv4 Packet Headers

You can apply a rewrite rule to MPLS and IPv4 packet headers simultaneously. This allows you to initialize MPLS EXP and IP precedence bits at LSP ingress. You can configure different rewrite rules depending on whether the traffic is VPN or non-VPN.

Copyright © 2019, Juniper Networks, Inc.

The default MPLS EXP rewrite rules are shown in Table 53 on page 404.

Table 53: Default MPLS EXP Rewrite Rules

Forwarding Class	Loss Priority	MPLS EXP Rewrite Value
best-effort	**low**	000
best-effort	**high**	001
expedited-forwarding	**low**	010
expedited-forwarding	**high**	011
assured-forwarding	**low**	100
assured-forwarding	**high**	101
network-control	**low**	110
network-control	**high**	111

By default, IP precedence rewrite rules alter the first three bits on the type-of-service (ToS) byte while leaving the last three bits unchanged. This default behavior applies to rewrite rules you configure for MPLS packets with IPv4 payloads.

To override the default MPLS EXP rewrite table and rewrite MPLS and IPv4 packet headers simultaneously, include the **protocol** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules exp *rewrite-rule-name*]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules exp
    rewrite-rule-name]
protocol protocol-types;
```

The **protocol** statement defines the types of MPLS packets and packet headers to which the specified rewrite rule is applied. The MPLS packet can be a standard MPLS packet or an MPLS packet with an IPv4 payload. Specify the type of MPLS packet using the following options:

- **mpls**—Applies the rewrite rule to MPLS packets and writes the CoS value to MPLS headers.
- **mpls-inet-both**—Applies the rewrite rule to VPN MPLS packets with IPv4 payloads. On Juniper Networks M120 Multiservice Edge Routers, M320 Multiservice Edge Routers, and T Series Core Routers (except T4000 routers), writes the CoS value to the MPLS and IPv4 headers. On other M Series Multiservice Edge Router routers, causes all ingress MPLS LSP packets with IPv4 payloads to be initialized with **000** code points for the MPLS EXP value, and the configured rewrite code point for IP precedence.
- **mpls-inet-both-non-vpn**—Applies the rewrite rule to non-VPN MPLS packets with IPv4 payloads. On Juniper Networks M120 Multiservice Edge Routers, M320 Multiservice

Copyright © 2019, Juniper Networks, Inc.

Edge Routers, and T Series Core Routers, writes the CoS value to the MPLS and IPv4 headers. On other M Series Multiservice Edge Routers, causes all ingress MPLS LSP packets with IPv4 payloads to be initialized with **000** code points for the MPLS EXP value, and the configured rewrite code point for IP precedence.

On M120 routers, M320 routers with Enhanced-III FPCs, and MX Series routers, you can perform simultaneous DSCP and EXP rewrite by attaching independent DSCP or IPv4 precedence rewrite rules and EXP rewrite rules to the same core interface. Thus, you can rewrite both code points (DSCP and EXP) when the packet is received by the ingress provider edge (PE) router on the MPLS core.

An alternative to overwriting the default with a rewrite-rules mapping is to configure the default packet header rewrite mappings, as discussed in "Applying Default Rewrite Rules" on page 389.

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

Example: Rewriting MPLS and IPv4 Packet Headers

On M320 and T Series routers, configure rewrite tables and apply them in various ways to achieve the following results:

- For interface **so-3/1/0**, the three EXP rewrite tables are applied to packets, depending on the protocol of the payload:
  - IPv4 packets (VPN) that enter the LSPs on interface **so-3/1/0** are initialized with values from rewrite table **exp-inet-table**. An identical 3-bit value is written into the IP precedence and MPLS EXP bit fields.
  - IPv4 packets (non-VPN) that enter the LSPs on interface **so-3/1/0** are initialized with values from rewrite table **rule-non-vpn**. An identical 3-bit value is written into the IP precedence and MPLS EXP bit fields.
  - Non-IPv4 packets that enter the LSPs on interface **so-3/1/0** are initialized with values from rewrite table **rule1**, and written into the MPLS EXP header field only. The statement **exp rule1** has the same result as **exp rule1 protocol mpls**.
- For interface **so-3/1/0**, IPv4 packets transmitted over a non-LSP layer are initialized with values from IP precedence rewrite table **rule2**.
- For interface **so-3/1/1**, IPv4 packets that enter the LSPs are initialized with values from EXP rewrite table **exp-inet-table**. An identical 3-bit value is written into the IP precedence and MPLS EXP bit fields.
- For interface **so-3/1/1**, MPLS packets other than IPv4 Layer 3 types are also initialized with values from table **exp-inet-table**. For VPN MPLS packets with IPv4 payloads, the

Copyright © 2019, Juniper Networks, Inc.

CoS value is written to MPLS and IPv4 headers. For VPN MPLS packets without IPv4 payloads, the CoS value is written to MPLS headers only.

```
[edit class-of-service]
rewrite-rules {
   exp exp-inet-table {
      forwarding-class best-effort {
         loss-priority low code-point 000;
         loss-priority high code-point 001;
      }
      forwarding-class assured-forwarding {
         loss-priority low code-point 010;
         loss-priority high code-point 011;
      }
      forwarding-class expedited-forwarding {
         loss-priority low code-point 111;
         loss-priority high code-point 110;
      }
      forwarding-class network-control {
         loss-priority low code-point 100;
         loss-priority high code-point 101;
      }
   }
   exp rule1 {
      ...
   }
   inet-precedence rule2 {
      ...
   }
}
exp rule_non_vpn {
   ...
}

interfaces {
   so-3/1/0 {
      unit 0 {
         rewrite-rules {
            exp rule1;
            inet-precedence rule2;
            exp exp-inet-table protocol mpls-inet-both; # For all VPN traffic.
            exp rule_non_vpn protocol mpls-inet-both-non-vpn; # For all non-VPN
               # traffic.
         }
      }
   }
   so-3/1/1 {
      unit 0 {
         rewrite-rules {
            exp exp-inet-table protocol [mpls mpls-inet-both];
         }
      }
   }
}
```


Copyright © 2019, Juniper Networks, Inc.

Example: Simultaneous DSCP and EXP Rewrite

On M120 routers, M320 routers with Enhanced-III FPCs, and MX Series routers, configure the simultaneous DSCP and EXP rewrite rules as shown below:

1. Configure CoS.

```
[edit]
user@host# edit class-of-service
```

2. Configure the EXP rewrite rule on the interface.

```
[edit class-of-service]
user@host# set interfaces ge-2/0/3 unit 0 rewrite-rule exp rule1
```

3. Configure the IPv4 rewrite rule on the interface.

```
[edit class-of-service]
user@host# set interfaces ge-2/0/3 unit 0 rewrite-rule inet-precedence rule2
```

4. Configure the IPv4 rewrite rule on the interface and apply it to packets entering the MPLS tunnel.

```
[edit class-of-service]
user@host# set interfaces ge-2/0/3 unit 0 rewrite-rule inet-precedence rule3 protocol
   mpls
```

5. Verify the configuration by using the **show interfaces** command.

```
[edit class-of-service]
user@host# show interfaces ge-2/0/3 unit 0
rewrite-rules {
exp rule1;
inet-precedence rule2;
inet-precedence rule3 protocol mpls;
}
```

In the example above, there are two different IPv4 precedence rewrite rules: **rule2** and **rule3**. **rule2** affects the IPv4 to IPv4 traffic and **rule3** affects the IPv4 to MPLS traffic.

Rewriting the EXP Bits of All Three Labels of an Outgoing Packet

In interprovider, carrier-of-carrier, and complex traffic engineering scenarios, it is sometimes necessary to push three labels on the next hop, using a swap-push-push or triple-push operation.

By default, on M Series routers and EX Series switches, the top MPLS EXP label of an outgoing packet is not rewritten when you configure swap-push-push and triple-push

Copyright © 2019, Juniper Networks, Inc.

operations. On M Series routers, you can rewrite the EXP bits of all three labels of an outgoing packet, thereby maintaining the CoS of an incoming MPLS or non-MPLS packet.

When the software performs a swap-push-push operation and no rewriting is configured, the EXP fields of all three labels are the same as in the old label. If there is EXP rewriting configured, the EXP bits of the bottom two labels are overwritten with the table entry. The EXP setting of the top label is retained even with rewriting.

To push three labels on all incoming MPLS packets, include the **exp-swap-push-push default** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]
exp-swap-push-push default;
```

When the software performs a push-push-push operation and if no rewriting is configured, the EXP fields of the bottom two labels are zero. If EXP rewriting is configured, the EXP fields of the bottom two labels are rewritten with the table entry's rewrite value. The EXP field of the top label is inserted with the Qn+PLP value. This Qn reflects the final classification by a multifield classifier if one exists, regardless of whether rewriting is configured.

NOTE: The **exp-push-push-push** and **exp-swap-push-push** configuration on the egress interface does not rewrite the top label's EXP field with the Qn+PLP value on an IQ or IQ2 PIC.

To push three labels on incoming non-MPLS packets, include the **exp-push-push-push default** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]
exp-push-push-push default;
```

These configurations apply the default MPLS EXP rewrite table, as described in "Rewriting MPLS and IPv4 Packet Headers" on page 403. You can configure these operations and override the default MPLS EXP rewrite table with a custom table. For more information about writing and applying a custom rewrite table, see "Configuring Rewrite Rules" on page 391 and "Applying Rewrite Rules to Output Logical Interfaces" on page 401.

NOTE: With a three-label stack, if you do not include the **exp-swap-push-push default** or **exp-push-push-push default** statement in the configuration, the top label's EXP bits are set to zero.

Example: Rewriting the EXP Bits of All Three Labels of an Outgoing Packet

Configure a swap-push-push operation, and override the default rewrite table with a custom table:

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service]
forwarding-classes {
   queue 0 be;
   queue 1 ef;
   queue 2 af;
   queue 3 nc;
}
interfaces {
   so-1/1/3 {
      unit 0 {
         rewrite-rules {
            exp exp_rew; # Apply custom rewrite table
            exp-swap-push-push default;
         }
      }
   }
}
rewrite-rules {
   exp exp_rew {
      forwarding-class be {
         loss-priority low code-point 000;
         loss-priority high code-point 100;
      }
      forwarding-class ef {
         loss-priority low code-point 001;
         loss-priority high code-point 101;
      }
      forwarding-class af {
         loss-priority low code-point 010;
         loss-priority high code-point 110;
      }
      forwarding-class nc {
         loss-priority low code-point 011;
         loss-priority high code-point 111;
      }
   }
}
```

Related Documentation

- Rewriting MPLS and IPv4 Packet Headers on page 403
- Configuring Rewrite Rules on page 391
- Applying Rewrite Rules to Output Logical Interfaces on page 401

Defining a Custom Frame Relay Loss Priority Map

You can apply a classifier to the same interface on which you configure a Frame Relay loss priority value. The Frame Relay loss priority map is applied first, followed by the classifier. The classifier can change the loss priority to a higher value only (for example, from low to high). If the classifier specifies a loss priority with a lower value than the current loss priority of a particular packet, the classifier does not change the loss priority of that packet.

Copyright © 2019, Juniper Networks, Inc.

To define a custom Frame Relay loss priority map:

1. At the **[edit class-of-service loss-priority-maps]** hierarchy level in configuration mode, specify the loss priority map for the Frame Relay DE bit.

   ```
   [edit class-of-service loss-priority-maps]
   user@host# set frame-relay-de name loss-priority level code-points [ alias | bits ];
   ```

   For example:

   ```
   [edit class-of-service loss-priority-maps]
   user@host# set frame-relay-de fr_rw loss-priority low code-points 0;
   user@host# set frame-relay-de fr_rw loss-priority high code-points 0;
   user@host# set frame-relay-de fr_rw loss-priority medium-low code-points 1;
   user@host# set frame-relay-de fr_rw loss-priority medium-high code-points 1;
   ```

   NOTE: The loss priority map does not take effect until you apply it to a logical interface.

2. Apply a rule to a logical interface.

   ```
   [edit class-of-service interfaces interface-name unit logical-unit-number
     loss-priority-maps]
   user@host# set frame-relay-de name;
   ```

   For example:

   ```
   [edit class-of-service interfaces so-1/0/0 unit 0 loss-priority-maps]
   user@host# set frame-relay-de fr_rw;
   ```

3. Verify the configuration in operational mode.

   ```
   user@host> show class-of-service loss-priority-map
   Loss-priority-map: frame-relay-de-fr_rw, Code point type: frame-relay-de,
   Index: 38
     Code point      Loss priority
     0                low
     0                high
     1                medium-low
     1                medium-high
   ```

Related Documentation

- frame-relay-de on page 1145

Example: Per-Node Rewriting of EXP Bits

To configure a custom table to rewrite the EXP bits, also known as CoS bits, on a particular node, the classifier table and the rewrite table must specify exactly the same CoS values.

Copyright © 2019, Juniper Networks, Inc.

In addition, the least significant bit of the CoS value itself must represent the PLP value. For example, CoS value **000** must be associated with PLP **low**, **001** must be associated with PLP **high**, and so forth.

This example configures a custom table to rewrite the EXP bits on a particular node:

```
[edit class-of-service]
classifiers {
   exp exp-class {
      forwarding-class be {
         loss-priority low code-points 000;
         loss-priority high code-points 001;
      }
      forwarding-class af {
         loss-priority low code-points 010;
         loss-priority high code-points 011;
      }
      forwarding-class ef {
         loss-priority low code-points 100;
         loss-priority high code-points 101;
      }
      forwarding-class nc {
         loss-priority low code-points 110;
         loss-priority high code-points 111;
      }
   }
}
rewrite-rules {
   exp exp-rw {
      forwarding-class be {
         loss-priority low code-point 000;
         loss-priority high code-point 001;
      }
      forwarding-class af {
         loss-priority low code-point 010;
         loss-priority high code-point 011;
      }
      forwarding-class ef {
         loss-priority low code-point 100;
         loss-priority high code-point 101;
      }
      forwarding-class nc {
         loss-priority low code-point 110;
         loss-priority high code-point 111;
      }
   }
}
```

Example: Rewriting CoS Information at the Network Border to Enforce CoS Strategies

This example shows how to rewrite (remark) class-of-service (CoS) values at the network border to enforce your internal CoS strategies. This is typically done when the CoS values of the inbound traffic at the network border cannot be trusted, or the values do not match your internal network's CoS strategy.

Copyright © 2019, Juniper Networks, Inc.

A thorough explanation of the CoS rewriting and its underlying algorithms is beyond the scope of this document. For more information about traffic policing, and CoS in general, refer to *QOS-Enabled Networks—Tools and Foundations* by Miguel Barreiros and Peter Lundqvist. This book is available at many online booksellers and at www.juniper.net/books .

- Requirements on page 412
- Overview on page 412
- Configuration on page 413
- Verification on page 420

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Overview

The purpose of this example is to demonstrate CoS rewriting at a network border to convey the traffics's CoS profile to the next-hop router, based on the forwarding class and packet loss priority (PLP) assigned to the traffic. CoS information rewriting is performed as the last step before a packet is transmitted onto the egress network.

In this example the rewriting is done when sending traffic from the host connected to Device R1 to the host connected to Device R2. The information required for rewriting the CoS parameters in the other direction is not included in this example. However, you can use the rewriting information in Device R1 (making changes for the interfaces used) and apply it to Device R2 to achieve bidirectional CoS rewriting.

Junos OS contains several default rewrite rules that might meet your requirements. You display them with the **show class-of-service rewrite-rule** command. A partial table of the default rewrite rule mappings is shown in the following table.

Map from Forwarding Class	PLP Value	MAP to DSCP/DSCP IPv6/EXP/IP Code Point Aliases
expedited-forwarding	low	ef
expedited-forwarding	high	ef
assured-forwarding	low	af11
assured-forwarding	high	af12(DSCP/DSCP IPv6/EXP)
best-effort	low	be
best-effort	high	be

Copyright © 2019, Juniper Networks, Inc.

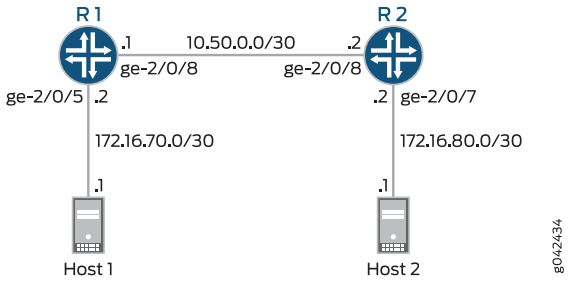
Map from Forwarding Class	PLP Value	MAP to DSCP/DSCP IPv6/EXP/IP Code Point Aliases
network-control	low	nc1/cs6
network-control	high	nc2/cs7

You can also define your own custom rewrite-rules table, or use a mixture of the default rewrite-rules and a custom table that you create. This example uses default rewrite-rules.

Topology

This example uses the topology in Figure 41 on page 413.

Figure 41: Rewriting CoS Information at the Network Border to Enforce CoS Strategies Scenario



This video explains the topics used in this example. We recommend that you watch the video before proceeding.

Video: Learning Bytes CoS Remarking Video.

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.

Device R1

```
set interfaces ge-2/0/5 description to-Host
set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
set interfaces ge-2/0/5 unit 0 family inet filter input mf-classifier
set interfaces ge-2/0/8 description to-R2
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
set interfaces lo0 unit 0 description looback-interface
set interfaces lo0 unit 0 family inet address 192.168.13.1/32
set class-of-service forwarding-classes queue 0 BE-data
set class-of-service forwarding-classes queue 1 Premium-data
```

Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service forwarding-classes queue 2 voice
set class-of-service forwarding-classes queue 3 NC
set class-of-service interfaces ge-2/0/8 scheduler-map test-map
set class-of-service interfaces ge-2/0/8 unit 0 rewrite-rules dscp IPv4-rewrite-table
set class-of-service rewrite-rules dscp IPv4-rewrite-table forwarding-class BE-data
  loss-priority low code-point be
set class-of-service rewrite-rules dscp IPv4-rewrite-table forwarding-class Premium-data
  loss-priority low code-point ef
set class-of-service scheduler-maps test-map forwarding-class BE-data scheduler
  BE-data
set class-of-service scheduler-maps test-map forwarding-class Premium-data scheduler
  Prem-data
set class-of-service schedulers BE-data transmit-rate 1m
set class-of-service schedulers BE-data buffer-size percent 25
set class-of-service schedulers BE-data priority low
set class-of-service schedulers Prem-data transmit-rate 1m
set class-of-service schedulers Prem-data buffer-size percent 25
set class-of-service schedulers Prem-data priority high
set firewall family inet filter mf-classifier term BE-data from protocol tcp
set firewall family inet filter mf-classifier term BE-data from port 80
set firewall family inet filter mf-classifier term BE-data then count BE-data
set firewall family inet filter mf-classifier term BE-data then forwarding-class BE-data
set firewall family inet filter mf-classifier term Prem-data from protocol tcp
set firewall family inet filter mf-classifier term Prem-data from port 12345
set firewall family inet filter mf-classifier term Prem-data then count Prem-data
set firewall family inet filter mf-classifier term Prem-data then forwarding-class
  Premium-data
set firewall family inet filter mf-classifier term accept then accept
set protocols ospf area 0.0.0.0 interface ge-2/0/5.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```

Device R2

```
set interfaces ge-2/0/7 description to-Host
set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.1/30
set interfaces ge-2/0/8 description to-R1
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set interfaces ge-2/0/8 unit 0 family inet filter input mf-classifier
set interfaces unit 0 description looback-interface
set interfaces unit 0 family inet address 192.168.14.1/32
set firewall family inet filter mf-classifier term BE-data from dscp be
set firewall family inet filter mf-classifier term BE-data then count BE-data
set firewall family inet filter mf-classifier term Premium-data from dscp ef
set firewall family inet filter mf-classifier term Premium-data then count Premium-data
set firewall family inet filter mf-classifier term accept then accept
set protocols ospf area 0.0.0.0 interface ge-2/0/7.0 passive
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
```


Copyright © 2019, Juniper Networks, Inc.

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Device R1:

1. Configure the device interfaces.

```
[edit ]
user@R1# set interfaces ge-2/0/5 description to-Host
user@R1# set interfaces ge-2/0/5 unit 0 family inet address 172.16.70.2/30
user@R1# set interfaces ge-2/0/5 unit 0 family inet filter input mf-classifier

user@R1# set interfaces ge-2/0/8 description to-R2
user@R1# set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30

user@R1# set interfaces lo0 unit 0 description looback-interface
user@R1# set interfaces lo0 unit 0 family inet address 192.168.13.1/32
```

2. Configure the firewall parameters.

```
[edit ]
user@R1# set firewall family inet filter mf-classifier term BE-data from protocol tcp
user@R1# set firewall family inet filter mf-classifier term BE-data from port 80
user@R1# set firewall family inet filter mf-classifier term BE-data then count BE-data
user@R1# set firewall family inet filter mf-classifier term BE-data then
  forwarding-class BE-data
user@R1# set firewall family inet filter mf-classifier term Prem-data from protocol
  tcp
user@R1# set firewall family inet filter mf-classifier term Prem-data from port 12345
user@R1# set firewall family inet filter mf-classifier term Prem-data then count
  Prem-data
user@R1# set firewall family inet filter mf-classifier term Prem-data then
  forwarding-class Premium-data
user@R1# set firewall family inet filter mf-classifier term accept then accept
```

3. Configure the class-of-service parameters.

```
[edit ]
user@R1# set class-of-service forwarding-classes queue 0 BE-data
user@R1# set class-of-service forwarding-classes queue 1 Premium-data
user@R1# set class-of-service forwarding-classes queue 2 voice
user@R1# set class-of-service forwarding-classes queue 3 NC

user@R1# set class-of-service interfaces ge-2/0/8 scheduler-map test-map

user@R1# set class-of-service interfaces ge-2/0/8 unit 0 rewrite-rules dscp
  IPv4-rewrite-table

user@R1# set class-of-service rewrite-rules dscp IPv4-rewrite-table forwarding-class
  BE-data loss-priority low code-point be
```

Copyright © 2019, Juniper Networks, Inc.

```
user@R1# set class-of-service rewrite-rules dscp IPv4-rewrite-table forwarding-class
  Premium-data loss-priority low code-point ef

user@R1# set class-of-service scheduler-maps test-map forwarding-class BE-data
  scheduler BE-data
user@R1# set class-of-service scheduler-maps test-map forwarding-class
  Premium-data scheduler Prem-data

user@R1# set class-of-service schedulers BE-data transmit-rate 1m
user@R1# set class-of-service schedulers BE-data buffer-size percent 25
user@R1# set class-of-service schedulers BE-data priority low
user@R1# set class-of-service schedulers Prem-data transmit-rate 1m
user@R1# set class-of-service schedulers Prem-data buffer-size percent 25
user@R1# set class-of-service schedulers Prem-data priority high
```

4. Configure OSPF.

```
[edit protocols ospf]
user@R1# set area 0.0.0.0 interface ge-2/0/5.0 passive
user@R1# set area 0.0.0.0 interface lo0.0 passive
user@R1# set area 0.0.0.0 interface ge-2/0/8.0
```

Step-by-Step Procedure

To configure Device R2:

1. Configure the device interface.

```
[edit ]
user@R1# set interfaces ge-2/0/7 description to-Host
user@R1# set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.1/30

user@R1# set interfaces ge-2/0/8 description to-R1
user@R1# set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
user@R2# set interfaces ge-2/0/8 unit 0 family inet filter input mf-classifier

user@R1# set interfaces unit 0 description looback-interface
user@R1# set interfaces unit 0 family inet address 192.168.14.1/32
```

2. Configure the firewall parameters.

```
[edit ]
user@R2# set firewall family inet filter mf-classifier term BE-data from dscp be
user@R2# set firewall family inet filter mf-classifier term BE-data then count BE-data
user@R2# set firewall family inet filter mf-classifier term Premium-data from dscp
  ef
user@R2# set firewall family inet filter mf-classifier term Premium-data then count
  Premium-data
user@R2# set firewall family inet filter mf-classifier term accept then accept
```


Copyright © 2019, Juniper Networks, Inc.

3. Configure OSPF.

```
[edit protocols ospf]
user@R1# set area 0.0.0.0 interface ge-2/0/7.0 passive
user@R1# set area 0.0.0.0 interface lo0.0 passive
user@R1# set area 0.0.0.0 interface ge-2/0/8.0
```

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

```
user@R1 show interfaces
   ge-2/0/5 {
   description to-Host;
   unit 0 {
      family inet {
         filter {
            input mf-classifier;
         }
         address 172.16.70.2/30;
      }
   }
}
ge-2/0/8 {
   description to-R2;
   unit 0 {
      family inet {
         address 10.50.0.1/30;
      }
   }
}
lo0 {
   unit 0 {
      description looback-interface;
      family inet {
         address 192.168.13.1/32;
      }
   }
}
```

```
user@R1 show firewall
family inet {
   filter mf-classifier {
      term BE-data {
         from {
            protocol tcp;
            port 80;
         }
         then {
            count BE-data;
```

Copyright © 2019, Juniper Networks, Inc.

```
            forwarding-class BE-data;
        }
      }
      term Prem-data {
        from {
          protocol tcp;
          port 12345;
        }
        then {
          count Prem-data;
          forwarding-class Premium-data;
        }
      }
      term accept {
        then accept;
      }
    }
}
```

```
user@R1 show class-of-service
forwarding-classes {
   queue 0 BE-data;
   queue 1 Premium-data;
   queue 2 voice;
   queue 3 NC;
}
interfaces {
   ge-2/0/8 {
      scheduler-map test-map;
      unit 0 {
         rewrite-rules {
            dscp IPv4-rewrite-table;
         }
      }
   }
}
rewrite-rules {
   dscp IPv4-rewrite-table {
      forwarding-class BE-data {
         loss-priority low code-point be;
      }
      forwarding-class Premium-data {
         loss-priority low code-point ef;
      }
   }
}
scheduler-maps {
   test-map {
      forwarding-class BE-data scheduler BE-data;
      forwarding-class Premium-data scheduler Prem-data;
   }
}
schedulers {
   BE-data {
```


Copyright © 2019, Juniper Networks, Inc.

```
    transmit-rate 1m;
    buffer-size percent 25;
    priority low;
  }
  Prem-data {
    transmit-rate 1m;
    buffer-size percent 25;
    priority high;
  }
}
```

```
user@R1# show protocols ospf
area 0.0.0.0 {
  interface ge-2/0/5.0 {
    passive;
  }
  interface lo0.0 {
    passive;
  }
  interface ge-2/0/8.0;
}
```

If you are done configuring Device R1, enter **commit** from configuration mode.

```
user@R2# show interfaces
ge-2/0/7 {
  unit 0 {
    description to-Host;
    family inet {
      address 172.16.80.2;
    }
  }
}
ge-2/0/8 {
  description to-R1;
  unit 0 {
    family inet {
      filter {
        input mf-classifier;
      }
      address 10.50.0.2/30;
    }
  }
}
lo0 {
  unit 0 {
    description looback-interface;
    family inet {
      address 192.168.14.1/32;
    }
  }
}
```

Copyright © 2019, Juniper Networks, Inc.

```
user@R2# show firewall
family inet {
   filter mf-classifier {
      term BE-data {
         from {
            dscp be;
         }
         then count BE-data;
      }
      term Premium-data {
         from {
            dscp ef;
         }
         then count Premium-data;
      }
      term accept {
         then accept;
      }
   }
}
```

```
user@R2# show protocols ospf
area 0.0.0.0 {
   interface ge-2/0/7.0 {
      passive;
   }
   interface lo0.0 {
      passive;
   }
   interface ge-2/0/8.0;
}
```

If you are done configuring Device R2, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Clearing the Firewall Counters on page 420
- Sending Traffic into the Network from TCP HTTP Ports 80 and 12345 and Monitoring the Results on page 421

Clearing the Firewall Counters

Purpose Confirm that the firewall counters are cleared.

Action On Devices R1 and R2, run the **clear firewall all** command to reset the firewall counters to 0.

```
user@R1> clear firewall all
user@R2> clear firewall all
```


Copyright © 2019, Juniper Networks, Inc.

Sending Traffic into the Network from TCP HTTP Ports 80 and 12345 and Monitoring the Results

Purpose Send traffic from the host connected to Device 1 into the network so that it can be monitored by the firewall on Device R1 and Device R2.

Action

1. Use a traffic generator to send 20 TCP packets with a source port of 80 into the network.

   The -s flag sets the source port. The -k flag causes the source port to remain steady at 80 instead of incrementing. The -c flag sets the number of packets to 20. The -d flag sets the packet size.

   ```
   [User@host]# hping 172.16.80.1 -c 20 -s 80 -k -d 300
   HPING 172.16.80.1 (eth1 172.16.80.1): NO FLAGS are set, 40 headers + 0 data
   bytes
   len=46 ip=172.16.80.1 ttl=62 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=0.9 ms
   .
   .
   .
   --- 172.16.80.1 hping statistic ---
   20 packets transmitted, 20 packets received, 0% packet loss
   round-trip min/avg/max = 0.9/9501.4/19002.4 ms
   ```

2. Use a traffic generator to send 20 TCP packets with a source port of 12345 into the network.

   ```
   [User@host]# hping 172.16.80.1 -c 20 -s 12345 -k -d 300
   HPING 172.16.80.1 (eth1 172.16.80.1): NO FLAGS are set, 40 headers + 0 data
   bytes
   len=46 ip=172.16.80.1 ttl=62 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=0.3 ms
   .
   .
   .
   --- 172.16.80.1 hping statistic ---
   20 packets transmitted, 20 packets received, 0% packet loss
   round-trip min/avg/max = 0.3/9501.5/19002.7 ms
   ```

3. On Device R1, check the firewall counters by using the **show firewall** command.

   ```
   user@R1> show firewall
   Filter: mf-classifier
   Counters:
   Name                                                Bytes              Packets
   BE-data                                               800                 20
   Prem-data                                             800                 20
   ```

4. On Device R2, check the firewall counters using the **show firewall** command.

   ```
   user@R2> show firewall
   ```

Copyright © 2019, Juniper Networks, Inc.

```
Filter: mf-classifier
Counters:
Name                                                Bytes              Packets
BE-data                                               800                 20
Premium-data                                          800                 20
```

Meaning Device R1 correctly set the code point for TCP packets to port 12345 to bf. Device R1 correctly set the code point for TCP packets to port 80 to ef. Device R2 correctly recognized the code point for TCP packets to port 12345 as bf. Device R2 correctly recognized the code point for TCP packets to port 80 as ef.

Related Documentation

- Example: Configuring and Applying Scheduler Maps on page 269

Example: Remarking Diffserv Code Points to MPLS EXPs to Carry CoS Profiles Across a Service Provider's L3VPN MPLS Network

This example is an introduction in how to rewrite (remark) DSCP class-of-service (CoS) code point values at the network border of a customer network and a service provider's MPLS network while maintaining the original CoS profile of the traffic so that the traffic can be remarked with the original DSCP code points when it exits the MPLS network.

- Requirements on page 422
- Overview on page 422
- Configuration on page 425
- Verification on page 444

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Overview

The purpose of rewriting the IP DSCP code point values to MPLS EXP code point values is to carry the packet's CoS profile across the service provider's MPLS network. The rewriting is performed by the provider edge (PE) routers at the borders of the service provider's network. See Figure 43 on page 424.

Junos OS contains several DSCP default rewrite rules that might meet your requirements. You display them with the **show class-of-service rewrite-rule** command. A partial set of the default rewrite DSCP code point rule mappings is shown in the following table.

Copyright © 2019, Juniper Networks, Inc.

You can also define your own custom rewrite-rules table, or use a mixture of the default rewrite-rules and a custom table that you create. This example uses default rewrite-rules.

Map from Forwarding Class	PLP Value	MAP to DSCP/DSCP IPv6/EXP/IP Code Point Aliases
expedited-forwarding	low	ef
expedited-forwarding	high	ef
assured-forwarding	low	af11
assured-forwarding	high	af12 (DSCP/DSCP IPv6/EXP)
best-effort	low	be
best-effort	high	be
network-control	low	nc1/cs6
network-control	high	nc2/cs7

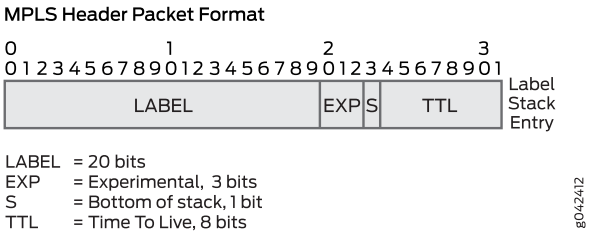
Junos OS uses the values shown in the following table for MPLS CoS in the EXP fields of the MPLS header.

Forwarding Class	Loss Priority	EXP Code Point
best-effort	low	000
best-effort	high	001
expedited-forwarding	low	010
expedited-forwarding	high	011
assured-forwarding	low	100
assured-forwarding	high	101
network-control	low	110
network-control	high	111

Figure 42 on page 424 shows the MPLS packet structure.

Copyright © 2019, Juniper Networks, Inc.

Figure 42: MPLS Packet Structure



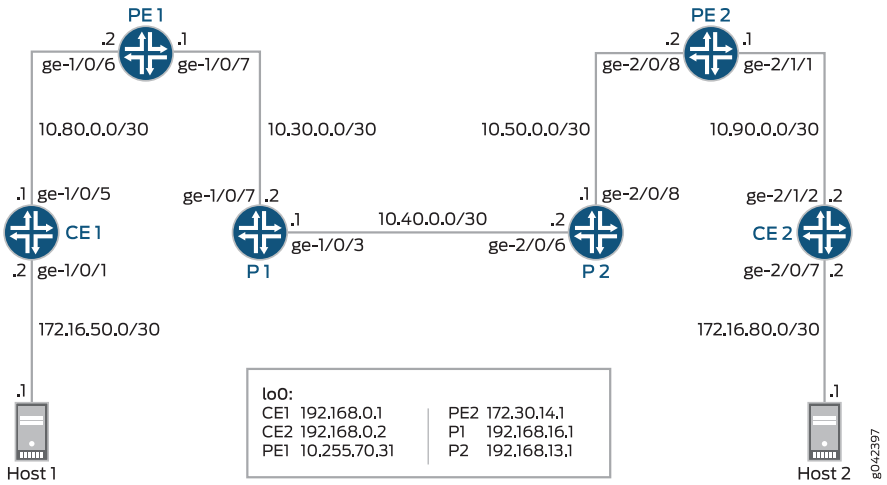
NOTE: In addition to providing the necessary information to complete the purpose of this example, this example also includes all of the commands required to re-create the Layer 3 VPN (L3VPN) network as shown in Figure 43 on page 424. A full explanation of the tasks required to configure an L3VPN network is not included in this example. If you require more information regarding configuring an L3VPN network, refer to the *Layer 3 VPNs Feature Guide for Routing Devices* available at http://juniper.net/documentation .

A thorough explanation of the required CoS rewriting and the underlying algorithms used in this example is beyond the scope of this document. For more information, refer to *QOS-Enabled Networks—Tools and Foundations* by Miguel Barreiros and Peter Lundqvist. This book is available at many online booksellers and at www.juniper.net/books .

Topology

This example uses the topology in Figure 43 on page 424.

Figure 43: Rewriting CoS Information at the Network Border to Transit an MPLS Network Scenario



Copyright © 2019, Juniper Networks, Inc.

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.

Device CE1

```
set interfaces ge-1/0/1 unit 0 description to-host
set interfaces ge-1/0/1 unit 0 family inet address 172.16.50.2/30
set interfaces ge-1/0/1 unit 0 family inet filter input ip-v4
set interfaces ge-1/0/5 unit 0 description to_Provider
set interfaces ge-1/0/5 unit 0 family inet address 10.80.0.1/30
set interfaces lo0 unit 1 description loopback-interface
set interfaces lo0 unit 1 family inet address 192.168.0.1/32
set protocols bgp group to_Provider type external
set protocols bgp group to_Provider export send-direct
set protocols bgp group to_Provider peer-as 64511
set protocols bgp group to_Provider neighbor 10.80.0.2
set policy-options policy-statement send-direct from protocol direct
set policy-options policy-statement send-direct then accept
set routing-options router-id 192.168.0.1
set routing-options autonomous-system 64510
set firewall family inet filter ip-v4 term tcp80 from port 80
set firewall family inet filter ip-v4 term tcp80 then dscp ef
set firewall family inet filter ip-v4 term 12345 from port 12345
set firewall family inet filter ip-v4 term 12345 then dscp be
set firewall family inet filter ip-v4 term accept then accept
```

Device PE1

```
set interfaces ge-1/0/6 description to_vpna
set interfaces ge-1/0/6 unit 0 family inet address 10.80.0.2/30
set interfaces ge-1/0/7 description to_P1
set interfaces ge-1/0/7 unit 0 family inet address 10.30.0.1/30
set interfaces ge-1/0/7 unit 0 family mpls
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 10.255.70.31/32
set routing-options router-id 10.255.70.31
set routing-options autonomous-system 64511
set protocols mpls interface ge-1/0/7.0
set protocols bgp group to_PE2 type internal
set protocols bgp group to_PE2 local-address 10.255.70.31
set protocols bgp group to_PE2 family inet-vpn unicast
set protocols bgp group to_PE2 neighbor 172.30.14.1
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-1/0/7.0
set protocols ldp interface ge-1/0/7.0
set protocols ldp interface lo0.0
set routing-instances vpna instance-type vrf
set routing-instances vpna interface ge-1/0/6.0
set routing-instances vpna route-distinguisher 64511:1
set routing-instances vpna vrf-target target:64511:1
```

Copyright © 2019, Juniper Networks, Inc.

```
set routing-instances vpna protocols bgp group to_vpna type external
set routing-instances vpna protocols bgp group to_vpna peer-as 64510
set routing-instances vpna protocols bgp group to_vpna neighbor 10.80.0.1
set class-of-service classifiers dscp dscpv4 forwarding-class expedited-forwarding
  loss-priority low code-points ef
set class-of-service classifiers dscp dscpv4 forwarding-class best-effort loss-priority low
  code-points be
set class-of-service classifiers exp exp-in forwarding-class expedited-forwarding
  loss-priority low code-points 010
set class-of-service classifiers exp exp-in forwarding-class best-effort loss-priority low
  code-points 000
set class-of-service interfaces ge-1/0/6 unit 0 classifiers dscp dscpv4
set class-of-service interfaces ge-1/0/6 unit 0 rewrite-rules dscp dscpv4-rw
set class-of-service interfaces ge-1/0/7 unit 0 classifiers exp exp-in
set class-of-service interfaces ge-1/0/7 unit 0 rewrite-rules exp exp-out
set class-of-service rewrite-rules dscp dscpv4-rw forwarding-class expedited-forwarding
  loss-priority low code-point ef
set class-of-service rewrite-rules dscp dscpv4-rw forwarding-class best-effort loss-priority
  low code-point be
set class-of-service rewrite-rules exp exp-out forwarding-class expedited-forwarding
  loss-priority low code-point 010
set class-of-service rewrite-rules exp exp-out forwarding-class best-effort loss-priority
  low code-point 000
```

Device P1

```
set interfaces ge-1/0/3 description to_P2
set interfaces ge-1/0/3 unit 0 family inet address 10.40.0.1/30
set interfaces ge-1/0/3 unit 0 family mpls
set interfaces ge-1/0/7 description to_PE1
set interfaces ge-1/0/7 unit 0 family inet address 10.30.0.2/30
set interfaces ge-1/0/7 unit 0 family mpls
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 192.168.16.1/32
set routing-options router-id 10.255.187.32
set protocols mpls interface ge-1/0/7.0
set protocols mpls interface ge-1/0/3.0
set protocols ospf area 0.0.0.0 interface ge-1/0/3.0
set protocols ospf area 0.0.0.0 interface ge-1/0/7.0
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ldp interface ge-1/0/3.0
set protocols ldp interface ge-1/0/7.0
set protocols ldp interface lo0.0
```

Device P2

```
set interfaces ge-2/0/6 description to_P1
set interfaces ge-2/0/6 unit 0 family inet address 10.40.0.2/30
set interfaces ge-2/0/6 unit 0 family mpls
set interfaces ge-2/0/8 description to_PE2
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
set interfaces ge-2/0/8 unit 0 family mpls
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 192.168.13.1/32
set routing-options router-id 192.168.187.3
set protocols mpls interface ge-2/0/6.0
```


Copyright © 2019, Juniper Networks, Inc.

```
set protocols mpls interface ge-2/0/8.0
set protocols ospf area 0.0.0.0 interface ge-2/0/6.0
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ldp interface ge-2/0/6.0
set protocols ldp interface ge-2/0/8.0
set protocols ldp interface lo0.0
```

Device PE2

```
set interfaces ge-2/0/8 description to-R1
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set interfaces ge-2/0/8 unit 0 family mpls
set interfaces ge-2/1/1 unit 0 description to-vpna
set interfaces ge-2/1/1 unit 0 family inet address 10.90.0.1/30
set interfaces ge-2/1/7 unit 0 family inet address 10.0.31.2/30
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 172.30.14.1
set routing-options router-id 172.30.14.1
set routing-options autonomous-system 64511
set protocols mpls interface ge-2/0/8.0
set protocols bgp group to_PE2 type internal
set protocols bgp group to_PE2 local-address 172.30.14.1
set protocols bgp group to_PE2 family inet-vpn unicast
set protocols bgp group to_PE2 neighbor 10.255.70.31
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ldp interface ge-2/0/8.0
set protocols ldp interface lo0.0
set routing-instances vpna instance-type vrf
set routing-instances vpna interface ge-2/1/1.0
set routing-instances vpna route-distinguisher 64511:1
set routing-instances vpna vrf-target target:64511:1
set routing-instances vpna protocols bgp group to_vpna type external
set routing-instances vpna protocols bgp group to_vpna peer-as 64512
set routing-instances vpna protocols bgp group to_vpna neighbor 10.90.0.2
set class-of-service classifiers dscp dscpv4 forwarding-class expedited-forwarding
  loss-priority low code-points ef
set class-of-service classifiers dscp dscpv4 forwarding-class best-effort loss-priority low
  code-points be
set class-of-service classifiers exp exp-in forwarding-class expedited-forwarding
  loss-priority low code-points 010
set class-of-service classifiers exp exp-in forwarding-class best-effort loss-priority low
  code-points 000
set class-of-service interfaces ge-2/0/8 unit 0 classifiers exp exp-in
set class-of-service interfaces ge-2/0/8 unit 0 rewrite-rules exp exp-out
set class-of-service interfaces ge-2/1/1 unit 0 classifiers dscp dscpv4
set class-of-service interfaces ge-2/1/1 unit 0 rewrite-rules dscp dscpv4-rw
set class-of-service rewrite-rules dscp dscpv4-rw forwarding-class expedited-forwarding
  loss-priority low code-point ef
set class-of-service rewrite-rules dscp dscpv4-rw forwarding-class best-effort loss-priority
  low code-point be
set class-of-service rewrite-rules exp exp-out forwarding-class expedited-forwarding
  loss-priority low code-point 010
```

Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service rewrite-rules exp exp-out forwarding-class best-effort loss-priority
  low code-point 000
```

Device CE2

```
set interfaces ge-2/0/7 unit 0 description to-host
set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
set interfaces ge-2/0/7 unit 0 family inet filter input ip-v4
set interfaces ge-2/1/2 unit 0 description to-Provider
set interfaces ge-2/1/2 unit 0 family inet address 10.90.0.2/30
set interfaces lo0 unit 1 description loopback-interface
set interfaces lo0 unit 1 family inet address 192.168.0.2/32
set protocols bgp group to_Provider type external
set protocols bgp group to_Provider export send-direct
set protocols bgp group to_Provider peer-as 64511
set protocols bgp group to_Provider neighbor 10.90.0.1
set policy-options policy-statement send-direct from protocol direct
set policy-options policy-statement send-direct then accept
set routing-options router-id 192.168.0.2
set routing-options autonomous-system 64512
set firewall family inet filter ip-v4 term tcp80 from port 80
set firewall family inet filter ip-v4 term tcp80 then dscp ef
set firewall family inet filter ip-v4 term 12345 from port 12345
set firewall family inet filter ip-v4 term 12345 then dscp be
set firewall family inet filter ip-v4 term accept then accept
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Device CE1:

1. Configure the device interfaces.

```
[edit ]
user@CE1# set interfaces ge-1/0/1 unit 0 description to-host
user@CE1# set interfaces ge-1/0/1 unit 0 family inet address 172.16.50.2/30
user@CE1# set interfaces ge-1/0/1 unit 0 family inet filter input ip-v4

user@CE1# set interfaces ge-1/0/5 unit 0 description to_Provider
user@CE1# set interfaces ge-1/0/5 unit 0 family inet address 10.80.0.1/30

user@CE1# set interfaces lo0 unit 1 description loopback-interface
user@CE1# set interfaces lo0 unit 1 family inet address 192.168.0.1/32
```

2. Configure the BGP parameters

```
[edit ]
user@CE1# set protocols bgp group to_Provider type external
user@CE1# set protocols bgp group to_Provider export send-direct
user@CE1# set protocols bgp group to_Provider peer-as 64511
user@CE1# set protocols bgp group to_Provider neighbor 10.80.0.2
```


Copyright © 2019, Juniper Networks, Inc.

3. Configure the policy option parameters.

```
[edit ]
user@CE1# set policy-options policy-statement send-direct from protocol direct
user@CE1# set policy-options policy-statement send-direct then accept
```

4. Configure the routing option parameters.

```
[edit ]
user@CE1# set routing-options router-id 192.168.0.1
user@CE1# set routing-options autonomous-system 64510
```

5. Configure the DSCP code point rewrite parameters.

```
[edit ]
user@CE1# set firewall family inet filter ip-v4 term tcp80 from port 80
user@CE1# set firewall family inet filter ip-v4 term tcp80 then dscp ef
user@CE1# set firewall family inet filter ip-v4 term 12345 from port 12345
user@CE1# set firewall family inet filter ip-v4 term 12345 then dscp be
user@CE1# set firewall family inet filter ip-v4 term accept then accept
```

Step-by-Step Procedure

To configure Device PE1:

1. Configure the device interfaces.

```
[edit ]
user@PE1# set interfaces ge-1/0/6 description to_vpna
user@PE1# set interfaces ge-1/0/6 unit 0 family inet address 10.80.0.2/30

user@PE1# set interfaces ge-1/0/7 description to_P1
user@PE1# set interfaces ge-1/0/7 unit 0 family inet address 10.30.0.1/30
user@PE1# set interfaces ge-1/0/7 unit 0 family mpls

user@PE1# set interfaces lo0 unit 0 description loopback-interface
user@PE1# set interfaces lo0 unit 0 family inet address 10.255.70.31/32
```

2. Configure the routing option parameters.

```
[edit ]
user@PE1# set routing-options router-id 10.255.70.31
user@PE1# set routing-options autonomous-system 64511
```

3. Configure the protocol parameters.

```
user@PE1# set protocols mpls interface ge-1/0/7.0

user@PE1# set protocols bgp group to_PE2 type internal
```

Copyright © 2019, Juniper Networks, Inc.

```
user@PE1# set protocols bgp group to_PE2 local-address 10.255.70.31
user@PE1# set protocols bgp group to_PE2 family inet-vpn unicast
user@PE1# set protocols bgp group to_PE2 neighbor 172.30.14.1

user@PE1# set protocols ospf area 0.0.0.0 interface lo0.0 passive
user@PE1# set protocols ospf area 0.0.0.0 interface ge-1/0/7.0

user@PE1# set protocols ldp interface ge-1/0/7.0
user@PE1# set protocols ldp interface lo0.0
```

4. Configure the routing instance parameters.

```
[edit ]
user@PE1# set routing-instances vpna instance-type vrf
user@PE1# set routing-instances vpna interface ge-1/0/6.0
user@PE1# set routing-instances vpna route-distinguisher 64511:1
user@PE1# set routing-instances vpna vrf-target target:64511:1
user@PE1# set routing-instances vpna protocols bgp group to_vpna type external
user@PE1# set routing-instances vpna protocols bgp group to_vpna peer-as 64510
user@PE1# set routing-instances vpna protocols bgp group to_vpna neighbor
  10.80.0.1
```

5. Configure the class-of-service parameters that perform the DSCP code point to MPLS EXP rewriting.

```
user@PE1# set class-of-service classifiers dscp dscpv4 forwarding-class
  expedited-forwarding loss-priority low code-points ef
user@PE1# set class-of-service classifiers dscp dscpv4 forwarding-class best-effort
  loss-priority low code-points be
user@PE1# set class-of-service classifiers exp exp-in forwarding-class
  expedited-forwarding loss-priority low code-points 010
user@PE1# set class-of-service classifiers exp exp-in forwarding-class best-effort
  loss-priority low code-points 000
user@PE1# set class-of-service interfaces ge-1/0/6 unit 0 classifiers dscp dscpv4
user@PE1# set class-of-service interfaces ge-1/0/6 unit 0 rewrite-rules dscp
  dscpv4-rw
user@PE1# set class-of-service interfaces ge-1/0/7 unit 0 classifiers exp exp-in
user@PE1# set class-of-service interfaces ge-1/0/7 unit 0 rewrite-rules exp exp-out
user@PE1# set class-of-service rewrite-rules dscp dscpv4-rw forwarding-class
  expedited-forwarding loss-priority low code-point ef
user@PE1# set class-of-service rewrite-rules dscp dscpv4-rw forwarding-class
  best-effort loss-priority low code-point be
user@PE1# set class-of-service rewrite-rules exp exp-out forwarding-class
  expedited-forwarding loss-priority low code-point 010
user@PE1# set class-of-service rewrite-rules exp exp-out forwarding-class
  best-effort loss-priority low code-point 000
```

Step-by-Step Procedure

To configure Device P1:

1. Configure the device interfaces.

Copyright © 2019, Juniper Networks, Inc.

```
[edit ]
user@P1# set interfaces ge-1/0/3 description to_P2
user@P1# set interfaces ge-1/0/3 unit 0 family inet address 10.40.0.1/30
user@P1# set interfaces ge-1/0/3 unit 0 family mpls

user@P1# set interfaces ge-1/0/7 description to_PE1
user@P1# set interfaces ge-1/0/7 unit 0 family inet address 10.30.0.2/30
user@P1# set interfaces ge-1/0/7 unit 0 family mpls

user@P1# set interfaces lo0 unit 0 description loopback-interface
user@P1# set interfaces lo0 unit 0 family inet address 192.168.16.1/32
```

2. Configure the routing option parameters.

```
[edit ]
user@P1# set routing-options router-id 10.255.187.32
```

3. Configure the protocol parameters.

```
[edit ]
user@P1# set protocols mpls interface ge-1/0/7.0
user@P1# set protocols mpls interface ge-1/0/3.0

user@P1# set protocols ospf area 0.0.0.0 interface ge-1/0/3.0
user@P1# set protocols ospf area 0.0.0.0 interface ge-1/0/7.0
user@P1# set protocols ospf area 0.0.0.0 interface lo0.0 passive

user@P1# set protocols ldp interface ge-1/0/3.0
user@P1# set protocols ldp interface ge-1/0/7.0
user@P1# set protocols ldp interface lo0.0
```

Step-by-Step Procedure

To configure Device P2:

1. Configure the device interfaces.

```
[edit ]
user@P2# set interfaces ge-2/0/6 description to_P1
user@P2# set interfaces ge-2/0/6 unit 0 family inet address 10.40.0.2/30
user@P2# set interfaces ge-2/0/6 unit 0 family mpls

user@P2# set interfaces ge-2/0/8 description to_PE2
user@P2# set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
user@P2# set interfaces ge-2/0/8 unit 0 family mpls

user@P2# set interfaces lo0 unit 0 description loopback-interface
user@P2# set interfaces lo0 unit 0 family inet address 192.168.13.1/32
```

Copyright © 2019, Juniper Networks, Inc.

2. Configure the routing option parameters.

```
[edit ]
user@P2# set routing-options router-id 192.168.187.3
```

3. Configure the protocol parameters.

```
[edit ]
user@P2# set protocols mpls interface ge-2/0/6.0
user@P2# set protocols mpls interface ge-2/0/8.0

user@P2# set protocols ospf area 0.0.0.0 interface ge-2/0/6.0
user@P2# set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
user@P2# set protocols ospf area 0.0.0.0 interface lo0.0 passive

user@P2# set protocols ldp interface ge-2/0/6.0
user@P2# set protocols ldp interface ge-2/0/8.0
user@P2# set protocols ldp interface lo0.0
```

Step-by-Step Procedure

To configure Device PE2:

1. Configure the device interfaces.

```
[edit ]
user@PE2# set interfaces ge-2/0/8 description to-R1
user@PE2# set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
user@PE2# set interfaces ge-2/0/8 unit 0 family mpls

user@PE2# set interfaces ge-2/1/1 unit 0 description to-vpna
user@PE2# set interfaces ge-2/1/1 unit 0 family inet address 10.90.0.1/30

user@PE2# set interfaces lo0 unit 0 description loopback-interface
user@PE2# set interfaces lo0 unit 0 family inet address 172.30.14.1/32
```

2. Configure the routing option parameters.

```
[edit ]
user@PE2# set routing-options router-id 172.30.14.1
user@PE2# set routing-options autonomous-system 64511
```

3. Configure the protocol parameters.

```
[edit ]
user@PE2# set protocols mpls interface ge-2/0/8.0

user@PE2# set protocols bgp group to_PE2 type internal
user@PE2# set protocols bgp group to_PE2 local-address 172.30.14.1
user@PE2# set protocols bgp group to_PE2 family inet-vpn unicast
```


Copyright © 2019, Juniper Networks, Inc.

```
user@PE2# set protocols bgp group to_PE2 neighbor 10.255.70.31


user@PE2# set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
user@PE2# set protocols ospf area 0.0.0.0 interface lo0.0 passive
user@PE2# set protocols ldp interface ge-2/0/8.0
user@PE2# set protocols ldp interface lo0.0
```

4. Configure the routing instance parameters.

```
[edit ]
user@PE2# set routing-instances vpna instance-type vrf
user@PE2# set routing-instances vpna interface ge-2/1/1.0
user@PE2# set routing-instances vpna route-distinguisher 64511:1
user@PE2# set routing-instances vpna vrf-target target:64511:1
user@PE2# set routing-instances vpna protocols bgp group to_vpna type external
user@PE2# set routing-instances vpna protocols bgp group to_vpna peer-as 64512
user@PE2# set routing-instances vpna protocols bgp group to_vpna neighbor
  10.90.0.2
```

5. Configure the class-of-service parameters that perform the DSCP code point to MPLS EXP rewriting.

```
[edit ]
user@PE2# set class-of-service classifiers dscp dscpv4 forwarding-class
  expedited-forwarding loss-priority low code-points ef
user@PE2# set class-of-service classifiers dscp dscpv4 forwarding-class best-effort
  loss-priority low code-points be
user@PE2# set class-of-service classifiers exp exp-in forwarding-class
  expedited-forwarding loss-priority low code-points 010
user@PE2# set class-of-service classifiers exp exp-in forwarding-class best-effort
  loss-priority low code-points 000
user@PE2# set class-of-service interfaces ge-2/0/8 unit 0 classifiers exp exp-in
user@PE2# set class-of-service interfaces ge-2/0/8 unit 0 rewrite-rules exp exp-out
user@PE2# set class-of-service interfaces ge-2/1/1 unit 0 classifiers dscp dscpv4
user@PE2# set class-of-service interfaces ge-2/1/1 unit 0 rewrite-rules dscp
  dscpv4-rw
user@PE2# set class-of-service rewrite-rules dscp dscpv4-rw forwarding-class
  expedited-forwarding loss-priority low code-point ef
user@PE2# set class-of-service rewrite-rules dscp dscpv4-rw forwarding-class
  best-effort loss-priority low code-point be
user@PE2# set class-of-service rewrite-rules exp exp-out forwarding-class
  expedited-forwarding loss-priority low code-point 010
user@PE2# set class-of-service rewrite-rules exp exp-out forwarding-class
  best-effort loss-priority low code-point 000
```

Step-by-Step Procedure

To configure Device CE2:

1. Configure the device interfaces.

```
[edit ]
```

Copyright © 2019, Juniper Networks, Inc.

```
user@CE2# set interfaces ge-2/0/7 unit 0 description to-host
user@CE2# set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
user@CE2# set interfaces ge-2/0/7 unit 0 family inet filter input ip-v4

user@CE2# set interfaces ge-2/1/2 unit 0 description to-Provider
user@CE2# set interfaces ge-2/1/2 unit 0 family inet address 10.90.0.2/30

set interfaces lo0 unit 1 description loopback-interface
set interfaces lo0 unit 1 family inet address 192.168.0.2/32
```

2. Configure the protocol parameters.

```
[edit ]
user@CE2# set protocols bgp group to_Provider type external
user@CE2# set protocols bgp group to_Provider export send-direct
user@CE2# set protocols bgp group to_Provider peer-as 64511
user@CE2# set protocols bgp group to_Provider neighbor 10.90.0.1
```

3. Configure the policy option parameters.

```
[edit ]
user@CE2# set policy-options policy-statement send-direct from protocol direct
user@CE2# set policy-options policy-statement send-direct then accept
```

4. Configure the routing option parameters.

```
[edit ]
user@CE2# set routing-options router-id 192.168.0.2
user@CE2# set routing-options autonomous-system 64512
```

5. Configure the DSCP code point rewrite parameters.

```
[edit ]
user@CE2# set firewall family inet filter ip-v4 term tcp80 from port 80
user@CE2# set firewall family inet filter ip-v4 term tcp80 then dscp ef
user@CE2# set firewall family inet filter ip-v4 term 12345 from port 12345
user@CE2# set firewall family inet filter ip-v4 term 12345 then dscp be
user@CE2# set firewall family inet filter ip-v4 term accept then accept
```

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

```
user@CE1# show interfaces
```


Copyright © 2019, Juniper Networks, Inc.

```
ge-1/0/1 {
    unit 0 {
        description to-host;
        family inet {
            filter {
                input ip-v4;
            }
            address 172.16.50.2/30;
        }
    }
}
ge-1/0/5 {
    unit 0 {
        description to_Provider;
        family inet {
            address 10.80.0.1/30;
        }
    }
}
lo0 {
    unit 1 {
        description loopback-interface;
        family inet {
            address 192.168.0.1/32;
        }
    }
}
```

```
user@CE1# show protocols
bgp {
    group to_Provider {
        type external;
        export send-direct;
        peer-as 64511;
        neighbor 10.80.0.2;
    }
}
```

```
user@CE1# show policy-options
policy-statement send-direct {
    from protocol direct;
    then accept;
}
```

```
user@CE1# show routing-options
router-id 192.168.0.1;
autonomous-system 64510;
```

```
user@CE1# show firewall
family inet {
    filter ip-v4 {
        term tcp80 {
            from {
```

Copyright © 2019, Juniper Networks, Inc.

```
            port 80;
        }
        then dscp ef;
      }
      term 12345 {
        from {
          port 12345;
        }
        then dscp be;
      }
      term accept {
        then accept;
      }
    }
}
```

If you are done configuring Device CE1, enter **commit** from configuration mode.

```
user@PE1# show interfaces
ge-1/0/6 {
    description to_vpna;
    unit 0 {
        family inet {
            address 10.80.0.2/30;
        }
    }
}
ge-1/0/7 {
    description to_P1;
    unit 0 {
        family inet {
            address 10.30.0.1/30;
        }
        family mpls;
    }
}
lo0 {
    unit 0 {
        description loopback-interface;
        family inet {
            address 10.255.70.31/32;
        }
    }
}
```

```
user@PE1# show protocols
mpls {
    interface ge-1/0/7.0;
}
bgp {
    group to_PE2 {
        type internal;
        local-address 10.255.70.31;
        family inet-vpn {
```


Copyright © 2019, Juniper Networks, Inc.

```
        unicast;
      }
      neighbor 172.30.14.1;
    }
}
ospf {
    area 0.0.0.0 {
      interface lo0.0 {
        passive;
      }
      interface ge-1/0/7.0;
    }
}
ldp {
    interface ge-1/0/7.0;
    interface lo0.0;
}
```

```
user@PE1# show routing-options
router-id 10.255.70.31;
autonomous-system 64511;
```

```
user@PE1# show routing-instances
vpna {
    instance-type vrf;
    interface ge-1/0/6.0;
    route-distinguisher 64511:1;
    vrf-target target:64511:1;
    protocols {
        bgp {
            group to_vpna {
                type external;
                peer-as 64510;
                neighbor 10.80.0.1;
            }
        }
    }
}
```

```
user@PE1# show class-of-service
classifiers {
    dscp dscpv4 {
        forwarding-class expedited-forwarding {
            loss-priority low code-points ef;
        }
        forwarding-class best-effort {
            loss-priority low code-points be;
        }
    }
    exp exp-in {
        forwarding-class expedited-forwarding {
            loss-priority low code-points 010;
        }
```

Copyright © 2019, Juniper Networks, Inc.

```
      forwarding-class best-effort {
        loss-priority low code-points 000;
      }
    }
}
interfaces {
    ge-1/0/6 {
      unit 0 {
        classifiers {
          dscp dscpv4;
        }
        rewrite-rules {
          dscp dscpv4-rw;
        }
      }
    }
    ge-1/0/7 {
      unit 0 {
        classifiers {
          exp exp-in;
        }
        rewrite-rules {
          exp exp-out;
        }
      }
    }
}
rewrite-rules {
    dscp dscpv4-rw {
      forwarding-class expedited-forwarding {
        loss-priority low code-point ef;
      }
      forwarding-class best-effort {
        loss-priority low code-point be;
      }
    }
    exp exp-out {
      forwarding-class expedited-forwarding {
        loss-priority low code-point 010;
      }
      forwarding-class best-effort {
        loss-priority low code-point 000;
      }
    }
}
```

If you are done configuring Device PE1, enter **commit** from configuration mode.

```
user@P1# show interfaces
ge-1/0/3 {
    description to_P2;
    unit 0 {
      family inet {
        address 10.40.0.1/30;
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
    family mpls;
  }
}
ge-1/0/7 {
  description to_PE1;
  unit 0 {
    family inet {
      address 10.30.0.2/30;
    }
    family mpls;
  }
}
lo0 {
  unit 0 {
    description loopback-interface;
    family inet {
      address 192.168.16.1/32;
    }
  }
}
```

```
user@P1# show protocols
mpls {
  interface ge-1/0/7.0;
  interface ge-1/0/3.0;
}
ospf {
  area 0.0.0.0 {
    interface ge-1/0/3.0;
    interface ge-1/0/7.0;
    interface lo0.0 {
      passive;
    }
  }
}
ldp {
  interface ge-1/0/3.0;
  interface ge-1/0/7.0;
  interface lo0.0;
}
```

```
user@P1# show routing-options
router-id 10.255.187.32;
```

If you are done configuring Device P1, enter **commit** from configuration mode.

```
user@P2# show interfaces
ge-2/0/6 {
  description to_P1;
  unit 0 {
    family inet {
```

Copyright © 2019, Juniper Networks, Inc.

```
        address 10.40.0.2/30;
      }
      family mpls;
    }
  }
  ge-2/0/8 {
    description to_PE2;
    unit 0 {
      family inet {
        address 10.50.0.1/30;
      }
      family mpls;
    }
  }
  lo0 {
    unit 0 {
      description loopback-interface;
      family inet {
        address 192.168.13.1/32;
      }
    }
  }
```

```
user@P2# show protocols
mpls {
  interface ge-2/0/6.0;
  interface ge-2/0/8.0;
}
ospf {
  area 0.0.0.0 {
    interface ge-2/0/6.0;
    interface ge-2/0/8.0;
    interface lo0.0 {
      passive;
    }
  }
}
ldp {
  interface ge-2/0/6.0;
  interface ge-2/0/8.0;
  interface lo0.0;
}
```

```
user@P2# show routing-options
router-id 192.168.187.3;
```

If you are done configuring Device P2, enter **commit** from configuration mode.

```
user@PE2# show interfaces
```

```
ge-2/0/8 {
  description to-R1;
  unit 0 {
```


Copyright © 2019, Juniper Networks, Inc.

```
      family inet {
        address 10.50.0.2/30;
      }
      family mpls;
    }
}
ge-2/1/1 {
    unit 0 {
      description to-vpna;
      family inet {
        address 10.90.0.1/30;
      }
    }
}
lo0 {
    unit 0 {
      description loopback-interface;
      family inet {
        address 172.30.14.1/32;
      }
    }
}
```

```
user@PE2# show protocols
mpls {
    interface ge-2/0/8.0;
}
bgp {
    group to_PE1 {
      type internal;
      local-address 172.30.14.1;
      family inet-vpn {
        unicast;
      }
      neighbor 10.255.70.31;
    }
}
ospf {
    area 0.0.0.0 {
      interface ge-2/0/8.0;
      interface lo0.0 {
        passive;
      }
    }
}
ldp {
    interface ge-2/0/8.0;
    interface lo0.0;
}
```

```
user@PE2# show routing-options
router-id 172.30.14.1;
autonomous-system 64511;
```

Copyright © 2019, Juniper Networks, Inc.

```
user@PE2# show routing-instances
vpna {
   instance-type vrf;
   interface ge-2/1/1.0;
   route-distinguisher 64511:1;
   vrf-target target:64511:1;
   protocols {
      bgp {
         group to_vpna {
            type external;
            peer-as 64512;
            neighbor 10.90.0.2;
         }
      }
   }
}
```

```
user@PE2# show class-of-service
classifiers {
   dscp dscpv4 {
      forwarding-class expedited-forwarding {
         loss-priority low code-points ef;
      }
      forwarding-class best-effort {
         loss-priority low code-points be;
      }
   }
   exp exp-in {
      forwarding-class expedited-forwarding {
         loss-priority low code-points 010;
      }
      forwarding-class best-effort {
         loss-priority low code-points 000;
      }
   }
}
interfaces {
   ge-2/0/8 {
      unit 0 {
         classifiers {
            exp exp-in;
         }
         rewrite-rules {
            exp exp-out;
         }
      }
   }
   ge-2/1/1 {
      unit 0 {
         classifiers {
            dscp dscpv4;
         }
         rewrite-rules {
            dscp dscpv4-rw;
```


Copyright © 2019, Juniper Networks, Inc.

```
        }
      }
    }
}
rewrite-rules {
    dscp dscpv4-rw {
        forwarding-class expedited-forwarding {
            loss-priority low code-point ef;
        }
        forwarding-class best-effort {
            loss-priority low code-point be;
        }
    }
    exp exp-out {
        forwarding-class expedited-forwarding {
            loss-priority low code-point 010;
        }
        forwarding-class best-effort {
            loss-priority low code-point 000;
        }
    }
}
```

If you are done configuring Device PE2, enter **commit** from configuration mode.

```
user@CE2# show interfaces
ge-2/0/7 {
    unit 0 {
        description to-host;
        family inet {
            filter {
                input ip-v4;
            }
            address 172.16.80.2/30;
        }
    }
}
ge-2/1/2 {
    unit 0 {
        description to-Provider;
        family inet {
            address 10.90.0.2/30;
        }
    }
}
lo0 {
    unit 1 {
        description loopback-interface;
        family inet {
            address 192.168.0.2/32;
        }
    }
}
```

Copyright © 2019, Juniper Networks, Inc.

```
user@CE2# show protocols
bgp {
   group to_Provider {
      type external;
      export send-direct;
      peer-as 64511;
      neighbor 10.90.0.1;
   }
}
```

```
user@CE2# show policy-options
policy-statement send-direct {
   from protocol direct;
   then accept;
}
```

```
user@CE2# show routing-options
router-id 192.168.0.2;
autonomous-system 64512;
```

```
user@CE2# show firewall
family inet {
   filter ip-v4 {
      term tcp80 {
         from {
            port 80;
         }
         then dscp ef;
      }
      term 12345 {
         from {
            port 12345;
         }
         then dscp be;
      }
      term accept {
         then accept;
      }
   }
}
```

If you are done configuring Device CE2, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly by verifying that the DSCP code points are maintained from CE1 to CE2.

- Clearing the Firewall Counters on page 445
- Sending Traffic into the Network from TCP HTTP Ports 80 and 12345 and Monitoring the Results on page 445

Copyright © 2019, Juniper Networks, Inc.

Clearing the Firewall Counters

Purpose Confirm that the firewall counters are cleared.

Action On Device CE2, run the **clear firewall all** command to reset the firewall counters to 0.

```
user@CE2> clear firewall all
```

Sending Traffic into the Network from TCP HTTP Ports 80 and 12345 and Monitoring the Results

Purpose Send traffic into the network from the host connected to Device CE1 so that it that can be monitored at Device CE2.

Action A different firewall is required on interface ge-2/0/7 to count the traffic that is being transmitted outbound to the destination. The following commands apply the firewall filter that counts the marked traffic as it is transmitted to the destination.

NOTE: To capture traffic at Device CE1, apply this command **set interfaces ge-1/0/1 unit 0 family inet filter output count**, followed by the commands below.

NOTE: To capture traffic at Device CE2, apply this command **set interfaces ge-2/0/7 unit 0 family inet filter output count**, followed by the commands below.

```
set firewall family inet filter count term be from dscp be
set firewall family inet filter count term be then count be
set firewall family inet filter count term ef from dscp ef
set firewall family inet filter count term ef then count ef
set firewall family inet filter count term accept then accept
set interfaces ge-2/0/7 unit 0 family inet filter output count
```

When you are done testing, you can leave the counting filter in place, or remove it.

1. On host 1 use a traffic generator to send 20 TCP packets with a source port of 80 into the network, and repeat the task using a source port of 12345.

   ```
   [user@host]# hping 172.16.80.1 -s 80 -k -c 20
   [user@host]# hping 172.16.80.1 -s 12345 -k -c 20
   ```

2. On Device CE2, check the firewall counters by using the **show firewall** command.

   ```
   user@CE2> show firewall
   ```

Copyright © 2019, Juniper Networks, Inc.

```
Filter: __CE2/ip-v4

Filter: __CE2/count
Counters:
Name                                                Bytes              Packets
be                                                    800                 20
ef                                                    800                 20
```

Meaning The code point for TCP packets to port 12345 is maintained as be. The code point for TCP packets to port 80 is maintained as ef.

Related Documentation
- Example: Configuring and Applying Scheduler Maps on page 269

Example: Remarking Diffserv Code Points to 802.1P PCPs to Carry CoS Profiles Across a Service Provider's VPLS Network

This configuration example explains how to implement class-of-service (CoS) capabilities over a Virtual Private LAN Service (VPLS) network.

- Requirements on page 446
- Overview on page 446
- Configuration on page 449
- Verification on page 467

Requirements

To verify this procedure, this example uses a traffic generator. The traffic generator can be hardware-based or it can be software running on a server or host machine.

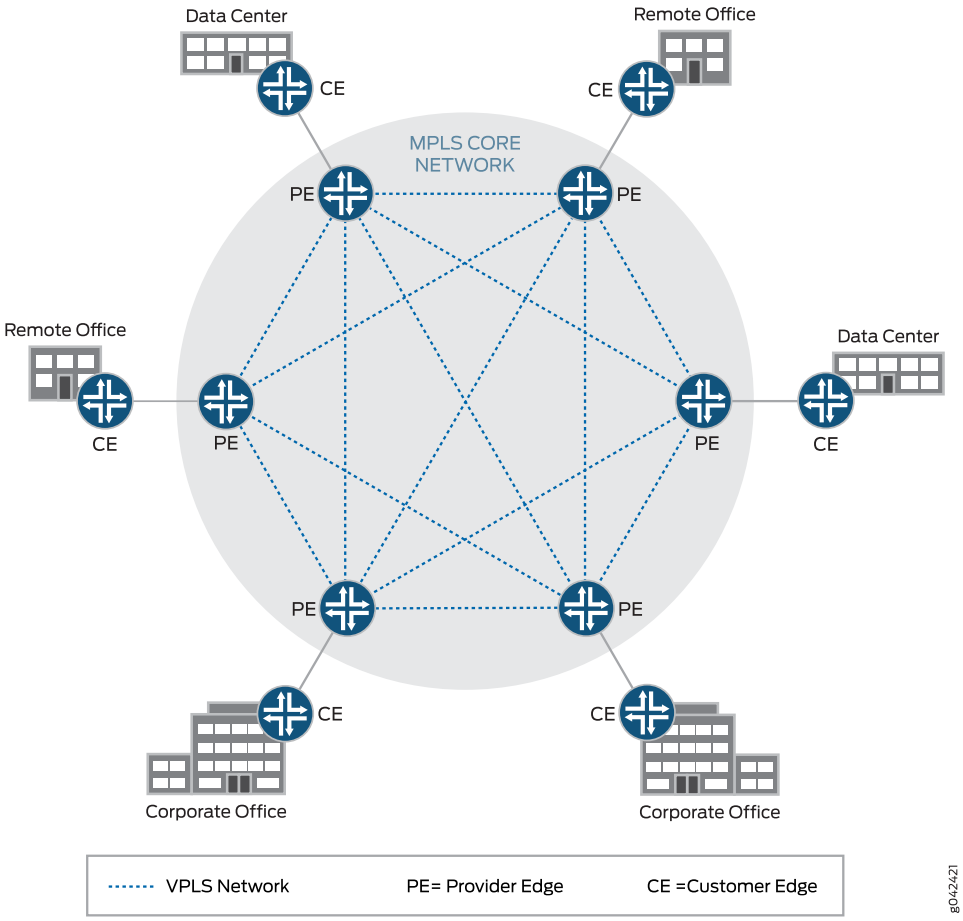
The functionality in this procedure is widely supported on devices that run Junos OS. The example shown here was tested and verified on MX Series routers running Junos OS Release 10.4.

Overview

VPLS networks create a Virtual Private LAN that provides a very close approximation of an Ethernet LAN to customers of a service provider. In a VPLS network, it is not necessary for all customers to be connected to a single LAN. Instead, the customers can be spread across two or more LANs. In the simplest sense, a VPLS network connects individual LANs across a packet-switched network so that they appear as a single LAN. See Figure 44 on page 447 for an example of a typical VPLS topology.

Copyright © 2019, Juniper Networks, Inc.

Figure 44: Typical VPLS Topology



Junos OS contains several DiffServ code point (DSCP) default rewrite rules that might meet your requirements. You display them with the **show class-of-service rewrite-rule** command. A partial set of the default rewrite DSCP rule mappings is shown in the following table.

You can also define your own custom rewrite-rules table, or use a mixture of the default rewrite-rules and a custom table that you create. This example uses default rewrite-rules.

Map from Forwarding Class	PLP Value	MAP to DSCP/DSCP IPv6/EXP/IP Code Point Aliases
expedited-forwarding	low	ef
expedited-forwarding	high	ef
assured-forwarding	low	af11
assured-forwarding	high	af12 (DSCP/DSCP IPv6/EXP)

Copyright © 2019, Juniper Networks, Inc.

Map from Forwarding Class	PLP Value	MAP to DSCP/DSCP IPv6/EXP/IP Code Point Aliases
best-effort	low	be
best-effort	high	be
network-control	low	nc1/cs6
network-control	high	nc2/cs7

Junos OS uses the values shown in the following table for MPLS CoS in the EXP fields of the MPLS header.

Forwarding Class	Loss Priority	EXP Code Point
best-effort	low	000
best-effort	high	001
expedited-forwarding	low	010
expedited-forwarding	high	011
assured-forwarding	low	100
assured-forwarding	high	101
network-control	low	110
network-control	high	111

NOTE: In addition to providing the necessary information to complete the purpose of this example, this example also includes all of the commands required to recreate the VPLS network as shown in Figure 45 on page 449. A full explanation of the tasks required to configure a VPLS network is not included in this example. If you need more information regarding configuring a VPLS network, see the *VPLS Feature Guide for Routing Devices* at http://juniper.net/documentation and RFC 4761 at http://tools.ietf.org/html/rfc4761 .

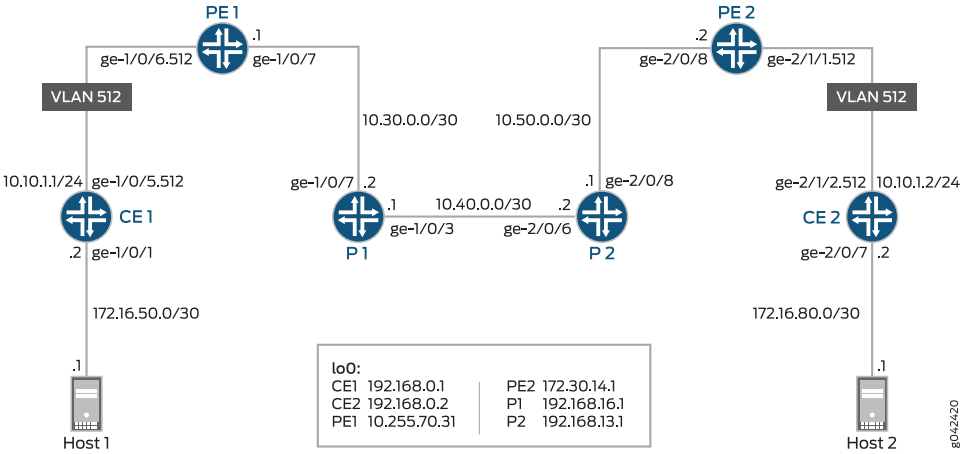
A thorough explanation of the required CoS tasks and the underlying algorithms used in this example is beyond the scope of this document. For more information, refer to *QOS-Enabled Networks—Tools and Foundations* by Miguel Barreiros and Peter Lundqvist. This book is available at many online booksellers and at www.juniper.net/books .

Copyright © 2019, Juniper Networks, Inc.

Topology

This example uses the topology in Figure 45 on page 449.

Figure 45: VPLS with CoS Scenario



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.

Device CE1

```
set interfaces ge-1/0/1 unit 0 description to-Host1
set interfaces ge-1/0/1 unit 0 family inet address 172.16.50.2/30
set interfaces ge-1/0/1 unit 0 family inet filter input ip-v4
set interfaces ge-1/0/5 vlan-tagging
set interfaces ge-1/0/5 unit 512 description to_PE1
set interfaces ge-1/0/5 unit 512 vlan-id 512
set interfaces ge-1/0/5 unit 512 family inet address 10.10.1.1/24
set interfaces lo0 unit 1 description loopback-interface
set interfaces lo0 unit 1 family inet address 192.168.0.1/32
set protocols ospf area 0.0.0.0 interface ge-1/0/5.512
set protocols ospf area 0.0.0.0 interface ge-1/0/1.0 passive
set protocols ospf area 0.0.0.0 interface lo0.1 passive
set firewall family inet filter ip-v4 term tcp80 from port 80
set firewall family inet filter ip-v4 term tcp80 then dscp ef
set firewall family inet filter ip-v4 term 12345 from port 12345
set firewall family inet filter ip-v4 term 12345 then dscp be
set firewall family inet filter ip-v4 term accept then accept
set class-of-service classifiers ieee-802.1 dscp1 forwarding-class expedited-forwarding
    loss-priority low code-points ef
set class-of-service classifiers ieee-802.1 dscp1 forwarding-class best-effort loss-priority
    low code-points be
set class-of-service rewrite-rules ieee-802.1 ieee1-c2 forwarding-class
    expedited-forwarding loss-priority low code-point 010
```

Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service rewrite-rules ieee-802.1 ieee1-c2 forwarding-class best-effort
  loss-priority low code-point 000
set class-of-service interfaces ge-1/0/5 unit 512 classifiers ieee-802.1 dscp1
set class-of-service interfaces ge-1/0/5 unit 512 rewrite-rules ieee-802.1 ieee1-c2
```

Device PE1

```
set interfaces ge-1/0/6 vlan-tagging
set interfaces ge-1/0/6 encapsulation vlan-vpls
set interfaces ge-1/0/6 unit 512 description to_vpls
set interfaces ge-1/0/6 unit 512 encapsulation vlan-vpls
set interfaces ge-1/0/6 unit 512 vlan-id 512
set interfaces ge-1/0/9 description to_P1
set interfaces ge-1/0/9 unit 0 family inet address 10.30.0.1/30
set interfaces ge-1/0/9 unit 0 family mpls
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 10.255.70.31/32
set protocols mpls interface ge-1/0/9.0
set protocols bgp group to_PE2 type internal
set protocols bgp group to_PE2 local-address 10.255.70.31
set protocols bgp group to_PE2 family l2vpn signaling
set protocols bgp group to_PE2 neighbor 172.30.14.1
set protocols ospf traffic-engineering
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ospf area 0.0.0.0 interface ge-1/0/9.0
set protocols ldp interface ge-1/0/9.0
set protocols ldp interface lo0.0
set routing-options router-id 10.255.70.31
set routing-options autonomous-system 64511
set routing-instances vpls_a instance-type vpls
set routing-instances vpls_a interface ge-1/0/6.512
set routing-instances vpls_a route-distinguisher 64511:1
set routing-instances vpls_a vrf-target target:64511:1
set routing-instances vpls_a protocols vpls no-tunnel-services
set routing-instances vpls_a protocols vpls site 1 site-identifier 1
set routing-instances vpls_a protocols vpls site 1 interface ge-1/0/6.512
```

Device P1

```
set interfaces ge-1/0/3 description to_P2
set interfaces ge-1/0/3 unit 0 family inet address 10.40.0.1/30
set interfaces ge-1/0/3 unit 0 family mpls
set interfaces ge-1/0/9 description to_PE1
set interfaces ge-1/0/9 unit 0 family inet address 10.30.0.2/30
set interfaces ge-1/0/9 unit 0 family mpls
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 192.168.16.1/32
set protocols mpls interface ge-1/0/9.0
set protocols mpls interface ge-1/0/3.0
set protocols ospf traffic-engineering
set protocols ospf area 0.0.0.0 interface ge-1/0/3.0
set protocols ospf area 0.0.0.0 interface ge-1/0/9.0
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ldp interface ge-1/0/3.0
set protocols ldp interface ge-1/0/9.0
set protocols ldp interface lo0.0
```


Copyright © 2019, Juniper Networks, Inc.

```
set routing-options router-id 192.168.16.1
```

Device P2

```
set interfaces ge-2/0/6 description to_P1
set interfaces ge-2/0/6 unit 0 family inet address 10.40.0.2/30
set interfaces ge-2/0/6 unit 0 family mpls
set interfaces ge-2/0/8 description to_PE2
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
set interfaces ge-2/0/8 unit 0 family mpls
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 192.168.13.1/32
set protocols mpls interface ge-2/0/6.0
set protocols mpls interface ge-2/0/8.0
set protocols ospf traffic-engineering
set protocols ospf area 0.0.0.0 interface ge-2/0/6.0
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ldp interface ge-2/0/6.0
set protocols ldp interface ge-2/0/8.0
set protocols ldp interface lo0.0
set routing-options router-id 192.168.13.1
```

Device PE2

```
set interfaces ge-2/0/8 description to-R1
set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
set interfaces ge-2/0/8 unit 0 family mpls
set interfaces ge-2/1/1 vlan-tagging
set interfaces ge-2/1/1 encapsulation vlan-vpls
set interfaces ge-2/1/1 unit 512 description to_vpls
set interfaces ge-2/1/1 unit 512 encapsulation vlan-vpls
set interfaces ge-2/1/1 unit 512 vlan-id 512
set interfaces lo0 unit 0 description loopback-interface
set interfaces lo0 unit 0 family inet address 172.30.14.1/32
set protocols mpls interface ge-2/0/8.0
set protocols bgp group to_PE1 type internal
set protocols bgp group to_PE1 local-address 172.30.14.1
set protocols bgp group to_PE1 family l2vpn signaling
set protocols bgp group to_PE1 neighbor 10.255.70.31
set protocols ospf traffic-engineering
set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
set protocols ospf area 0.0.0.0 interface lo0.0 passive
set protocols ldp interface ge-2/0/8.0
set protocols ldp interface lo0.0
set routing-options router-id 172.30.14.1
set routing-options autonomous-system 64511
set routing-instances vpls_a instance-type vpls
set routing-instances vpls_a interface ge-2/1/1.512
set routing-instances vpls_a route-distinguisher 64511:1
set routing-instances vpls_a vrf-target target:64511:1
set routing-instances vpls_a protocols vpls no-tunnel-services
set routing-instances vpls_a protocols vpls site 2 site-identifier 2
set routing-instances vpls_a protocols vpls site 2 interface ge-2/1/1.512
```

Copyright © 2019, Juniper Networks, Inc.

Device CE2

```
set interfaces ge-2/0/7 unit 0 description to-Host2
set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
set interfaces ge-2/0/7 unit 0 family inet filter input ip-v4
set interfaces ge-2/1/2 vlan-tagging
set interfaces ge-2/1/2 unit 512 description to-PE2
set interfaces ge-2/1/2 unit 512 vlan-id 512
set interfaces ge-2/1/2 unit 512 family inet address 10.10.1.2/24
set interfaces lo0 unit 1 description loopback-interface
set interfaces lo0 unit 1 family inet address 192.168.0.2/32
set protocols ospf area 0.0.0.0 interface lo0.1 passive
set protocols ospf area 0.0.0.0 interface ge-2/0/7.0 passive
set protocols ospf area 0.0.0.0 interface ge-2/1/2.512
set firewall family inet filter ip-v4 term tcp80 from port 80
set firewall family inet filter ip-v4 term tcp80 then dscp ef
set firewall family inet filter ip-v4 term 12345 from port 12345
set firewall family inet filter ip-v4 term 12345 then dscp be
set firewall family inet filter ip-v4 term accept then accept
set class-of-service classifiers ieee-802.1 dscp1 forwarding-class expedited-forwarding
  loss-priority low code-points ef
set class-of-service classifiers ieee-802.1 dscp1 forwarding-class best-effort loss-priority
  low code-points be
set class-of-service rewrite-rules ieee-802.1 ieee1-c2 forwarding-class
  expedited-forwarding loss-priority low code-point 010
set class-of-service rewrite-rules ieee-802.1 ieee1-c2 forwarding-class best-effort
  loss-priority low code-point 000
set class-of-service interfaces ge-2/1/2 unit 512 rewrite-rules ieee-802.1 ieee1-c2
set class-of-service interfaces ge-2/1/2 unit 512 classifiers ieee-802.1 dscp1
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure Device CE1:

1. Configure the device interfaces.

```
[edit ]
user@CE1# set interfaces ge-1/0/1 unit 0 description to-Host1
user@CE1# set interfaces ge-1/0/1 unit 0 family inet address 172.16.50.2/30
user@CE1#set interfaces ge-1/0/1 unit 0 family inet filter input ip-v4


user@CE1#set interfaces lo0 unit 1 description loopback-interface
user@CE1# set interfaces lo0 unit 1 family inet address 192.168.0.1/32
```

2. Configure the VLAN parameters.

```
[edit ]
user@CE1# set interfaces ge-1/0/5 vlan-tagging
user@CE1# set interfaces ge-1/0/5 unit 512 description to_PE1
user@CE1# set interfaces ge-1/0/5 unit 512 vlan-id 512
user@CE1# set interfaces ge-1/0/5 unit 512 family inet address 10.10.1.1/24
```


Copyright © 2019, Juniper Networks, Inc.

3. Configure the class-of-service parameters.

```
[edit ]
user@CE1# set class-of-service rewrite-rules ieee-802.1 ieee1-c2 forwarding-class
  expedited-forwarding loss-priority low code-point 010
user@CE1# set class-of-service rewrite-rules ieee-802.1 ieee1-c2 forwarding-class
  best-effort loss-priority low code-point 000
user@CE1# set class-of-service classifiers ieee-802.1 dscp1 forwarding-class
  expedited-forwarding loss-priority low code-points ef
user@CE1# set class-of-service classifiers ieee-802.1 dscp1 forwarding-class
  best-effort loss-priority low code-points be
user@CE1# set class-of-service interfaces ge-1/0/5 unit 512 rewrite-rules ieee-802.1
  ieee1-c2
user@CE1# set class-of-service interfaces ge-1/0/5 unit 512 classifiers ieee-802.1
  dscp1
```

4. Configure the protocol parameters.

```
[edit ]
user@CE1# set protocols ospf area 0.0.0.0 interface ge-1/0/5.512
user@CE1# set protocols ospf area 0.0.0.0 interface ge-1/0/1.0 passive
user@CE1# set protocols ospf area 0.0.0.0 interface lo0.0 passive
```

5. Configure the firewall DSCP rewrite parameters.

```
[edit ]
user@CE1# set firewall family inet filter ip-v4 term tcp80 from port 80
user@CE1# set firewall family inet filter ip-v4 term tcp80 then dscp ef
user@CE1# set firewall family inet filter ip-v4 term 12345 from port 12345
user@CE1# set firewall family inet filter ip-v4 term 12345 then dscp be
user@CE1# set firewall family inet filter ip-v4 term accept then accept
```

Step-by-Step Procedure

To configure Device PE1:

1. Configure the device interfaces.

```
[edit ]
user@PE1# set interfaces ge-1/0/9 description to_P1
user@PE1# set interfaces ge-1/0/9 unit 0 family inet address 10.30.0.1/30
user@PE1# set interfaces ge-1/0/9 unit 0 family mpls

user@PE1# set interfaces lo0 unit 0 description loopback-interface
user@PE1# set interfaces lo0 unit 0 family inet address 10.255.70.31/32
```

2. Configure the VLAN parameters.

```
[edit ]
user@PE1# set interfaces ge-1/0/6 vlan-tagging
user@PE1# set interfaces ge-1/0/6 encapsulation vlan-vpls
```

Copyright © 2019, Juniper Networks, Inc.

```
user@PE1# set interfaces ge-1/0/6 unit 512 description to_vpls
user@PE1# set interfaces ge-1/0/6 unit 512 encapsulation vlan-vpls
user@PE1# set interfaces ge-1/0/6 unit 512 vlan-id 512
```

3. Configure the protocol parameters.

```
[edit ]
user@PE1# set protocols mpls interface ge-1/0/9.0

user@PE1# set protocols bgp group to_PE2 type internal
user@PE1# set protocols bgp group to_PE2 local-address 10.255.70.31
user@PE1# set protocols bgp group to_PE2 family l2vpn signaling
user@PE1# set protocols bgp group to_PE2 neighbor 172.30.14.1

user@PE1# set protocols ospf traffic-engineering
user@PE1# set protocols ospf area 0.0.0.0 interface lo0.0 passive
user@PE1# set protocols ospf area 0.0.0.0 interface ge-1/0/9.0

user@PE1# set protocols ldp interface ge-1/0/9.0
user@PE1# set protocols ldp interface lo0.0
```

4. Configure the routing option parameters.

```
[edit ]
user@PE1# set routing-options router-id 10.255.70.31
user@PE1# set routing-options autonomous-system 64511
```

5. Configure the routing instance parameters.

```
[edit ]
user@PE1# set routing-instances vpls_a instance-type vpls
user@PE1# set routing-instances vpls_a interface ge-1/0/6.512
user@PE1# set routing-instances vpls_a route-distinguisher 64511:1
user@PE1# set routing-instances vpls_a vrf-target target:64511:1
user@PE1# set routing-instances vpls_a protocols vpls no-tunnel-services
user@PE1# set routing-instances vpls_a protocols vpls site 1 site-identifier 1
user@PE1# set routing-instances vpls_a protocols vpls site 1 interface ge-1/0/6.512
```

Step-by-Step Procedure

To configure Device P1:

1. Configure the device interfaces.

```
[edit ]
user@P1# set interfaces ge-1/0/3 description to_P2
user@P1# set interfaces ge-1/0/3 unit 0 family inet address 10.40.0.1/30
user@P1# set interfaces ge-1/0/3 unit 0 family mpls

user@P1# set interfaces ge-1/0/9 description to_PE1
```


Copyright © 2019, Juniper Networks, Inc.

```
user@P1# set interfaces ge-1/0/9 unit 0 family inet address 10.30.0.2/30
user@P1# set interfaces ge-1/0/9 unit 0 family mpls

user@P1# set interfaces lo0 unit 0 description loopback-interface
user@P1# set interfaces lo0 unit 0 family inet address 192.168.16.1/32
```

2. Configure the protocol parameters.

```
[edit ]
user@P1# set protocols mpls interface ge-1/0/9.0
user@P1# set protocols mpls interface ge-1/0/3.0

user@P1# set protocols ospf traffic-engineering
user@P1# set protocols ospf area 0.0.0.0 interface ge-1/0/3.0
user@P1# set protocols ospf area 0.0.0.0 interface ge-1/0/9.0
user@P1# set protocols ospf area 0.0.0.0 interface lo0.0 passive

user@P1# set protocols ldp interface ge-1/0/3.0
user@P1# set protocols ldp interface ge-1/0/9.0
user@P1# set protocols ldp interface lo0.0
```

3. Configure the routing options parameter.

```
[edit ]
user@P1# set routing-options router-id 192.168.16.1
```

Step-by-Step Procedure

To configure Device P2:

1. Configure the device interfaces.

```
[edit ]
user@P2# set interfaces ge-2/0/6 description to_P1
user@P2#set interfaces ge-2/0/6 unit 0 family inet address 10.40.0.2/30
user@P2# set interfaces ge-2/0/6 unit 0 family mpls

user@P2# set interfaces ge-2/0/8 description to_PE2
user@P2# set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.1/30
user@P2# set interfaces ge-2/0/8 unit 0 family mpls

user@P2# set interfaces lo0 unit 0 description loopback-interface
user@P2# set interfaces lo0 unit 0 family inet address 192.168.13.1/32
```

2. Configure the protocol parameters.

```
[edit ]
user@P2# set protocols mpls interface ge-2/0/6.0
user@P2# set protocols mpls interface ge-2/0/8.0
```

Copyright © 2019, Juniper Networks, Inc.

```
user@P2# set protocols ospf traffic-engineering
user@P2#  set protocols ospf area 0.0.0.0 interface ge-2/0/6.0
user@P2# set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
user@P2# set protocols ospf area 0.0.0.0 interface lo0.0 passive

user@P2# set protocols ldp interface ge-2/0/6.0
user@P2# set protocols ldp interface ge-2/0/8.0
user@P2# set protocols ldp interface lo0.0
```

3. Configure the routing option parameter.

```
[edit ]
user@P2# set routing-options router-id 192.168.13.1
```

Step-by-Step Procedure

To configure Device PE2:

1. Configure the device interfaces.

```
[edit ]
user@PE2# set interfaces ge-2/0/8 description to-R1
user@PE2# set interfaces ge-2/0/8 unit 0 family inet address 10.50.0.2/30
user@PE2# set interfaces ge-2/0/8 unit 0 family mpls

user@PE2# set interfaces lo0 unit 0 description loopback-interface
user@PE2# set interfaces lo0 unit 0 family inet address 172.30.14.1/32
```

2. Configure the VLAN parameters.

```
[edit ]
user@PE2# set interfaces ge-2/1/1 vlan-tagging
user@PE2# set interfaces ge-2/1/1 encapsulation vlan-vpls
user@PE2# set interfaces ge-2/1/1 unit 512 description to_vpls
user@PE2# set interfaces ge-2/1/1 unit 512 encapsulation vlan-vpls
user@PE2# set interfaces ge-2/1/1 unit 512 vlan-id 512
```

3. Configure the protocol parameters.

```
[edit ]
user@PE2# set protocols mpls interface ge-2/0/8.0

user@PE2# set protocols bgp group to_PE1 type internal
user@PE2# set protocols bgp group to_PE1 local-address 172.30.14.1
user@P2# set protocols bgp group to_PE1 family l2vpn signaling
user@PE2# set protocols bgp group to_PE1 neighbor 10.255.70.31

user@PE2# set protocols ospf traffic-engineering
user@PE2# set protocols ospf area 0.0.0.0 interface ge-2/0/8.0
user@PE2# set protocols ospf area 0.0.0.0 interface lo0.0 passive
```


Copyright © 2019, Juniper Networks, Inc.

```
user@PE2# set protocols ldp interface ge-2/0/8.0
user@PE2# set protocols ldp interface lo0.0
```

4. Configure the routing option parameters.

```
[edit ]
user@PE2# set routing-options router-id 172.30.14.1
user@PE2# set routing-options autonomous-system 64511
```

5. Configure the routing instance parameters.

```
[edit ]
user@P2# set routing-instances vpls_a instance-type vpls
user@PE2# set routing-instances vpls_a interface ge-2/1/1.512
user@PE2# set routing-instances vpls_a route-distinguisher 64511:1
user@PE2# set routing-instances vpls_a vrf-target target:64511:1
user@PE2# set routing-instances vpls_a protocols vpls no-tunnel-services
user@PE2# set routing-instances vpls_a protocols vpls site 2 site-identifier 2
user@PE2# set routing-instances vpls_a protocols vpls site 2 interface ge-2/1/1.512
```

Step-by-Step Procedure

To configure Device CE2:

1. Configure the device interfaces.

```
[edit ]
user@CE2# set interfaces ge-2/0/7 unit 0 description to-Host2
user@CE2# set interfaces ge-2/0/7 unit 0 family inet address 172.16.80.2/30
user@CE2# set interfaces ge-2/0/7 unit 0 family inet filter input ip-v4

user@CE2# set interfaces lo0 unit 1 description loopback-interface
user@CE2# set interfaces lo0 unit 1 family inet address 192.168.0.2/32
```

2. Configure the VLAN parameters

```
[edit ]
user@CE2# set interfaces ge-2/1/2 vlan-tagging
user@CE2# set interfaces ge-2/1/2 unit 512 description to-PE2
user@CE2# set interfaces ge-2/1/2 unit 512 vlan-id 512
user@CE2# set interfaces ge-2/1/2 unit 512 family inet address 10.10.1.2/24
```

3. Configure the class-of-service parameters.

```
[edit ]
user@CE2# set class-of-service rewrite-rules ieee-802.1 ieee1-c2 forwarding-class
  expedited-forwarding loss-priority low code-point 010
user@CE2# set class-of-service rewrite-rules ieee-802.1 ieee1-c2 forwarding-class
  best-effort loss-priority low code-point 000
```

Copyright © 2019, Juniper Networks, Inc.

```
user@CE2# set class-of-service classifiers ieee-802.1 dscp1 forwarding-class
  expedited-forwarding loss-priority low code-points ef
user@CE2# set class-of-service classifiers ieee-802.1 dscp1 forwarding-class
  best-effort loss-priority low code-points be
user@CE2# set class-of-service interfaces ge-2/1/2 unit 512 rewrite-rules ieee-802.1
  ieee1-c2
user@CE2# set class-of-service interfaces ge-2/1/2 unit 512 classifiers ieee-802.1
  dscp1
```

4. Configure the protocol parameters.

```
[edit ]
user@CE2# set protocols ospf area 0.0.0.0 interface lo0.0 passive
user@CE2# set protocols ospf area 0.0.0.0 interface ge-2/0/7.0 passive
user@CE2# set protocols ospf area 0.0.0.0 interface ge-2/1/2.512
```

5. Configure the firewall DSCP rewrite parameters.

```
[edit ]
user@CE2# set firewall family inet filter ip-v4 term tcp80 from port 80
user@CE2# set firewall family inet filter ip-v4 term tcp80 then dscp ef
user@CE2# set firewall family inet filter ip-v4 term 12345 from port 12345
user@CE2# set firewall family inet filter ip-v4 term 12345 then dscp be
user@CE2# set firewall family inet filter ip-v4 term accept then accept
```

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

```
user@CE1# show interfaces
ge-1/0/1 {
   unit 0 {
      description to-Host1;
      family inet {
         filter {
            input ip-v4;
         }
         address 172.16.50.2/30;
      }
   }
}
ge-1/0/5 {
   vlan-tagging;
   unit 512 {
      description to_PE1;
      vlan-id 512;
      family inet {
         address 10.10.1.1/24;
      }
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
}
lo0 {
    unit 1 {
        description loopback-interface;
        family inet {
            address 192.168.0.1/32;
        }
    }
}
```

```
user@CE1# show class-of-service
classifiers {
    ieee-802.1 dscp1 {
        forwarding-class expedited-forwarding {
            loss-priority low code-points ef;
        }
        forwarding-class best-effort {
            loss-priority low code-points be;
        }
    }
}
interfaces {
    ge-1/0/5 {
        unit 512 {
            classifiers {
                ieee-802.1 dscp1;
            }
            rewrite-rules {
                ieee-802.1 ieee1-c2;
            }
        }
    }
}
rewrite-rules {
    ieee-802.1 ieee1-c2 {
        forwarding-class expedited-forwarding {
            loss-priority low code-point 010;
        }
        forwarding-class best-effort {
            loss-priority low code-point 000;
        }
    }
}
```

```
user@CE1# show protocols
ospf {
    area 0.0.0.0 {
        interface ge-1/0/5.512;
        interface ge-1/0/1.0 {
            passive;
        }
        interface lo0.1 {
            passive;
```

Copyright © 2019, Juniper Networks, Inc.

```
    }
  }
}
```

```
user@CE1# show firewall
family inet {
  filter ip-v4 {
    term tcp80 {
      from {
        port 80;
      }
      then dscp ef;
    }
    term 12345 {
      from {
        port 12345;
      }
      then dscp be;
    }
    term accept {
      then accept;
    }
  }
}
```

If you are done configuring Device CE1, enter **commit** from configuration mode.

```
user@PE1# show interfaces
ge-1/0/6 {
  vlan-tagging;
  encapsulation vlan-vpls;
  unit 512 {
    description to_vpls;
    encapsulation vlan-vpls;
    vlan-id 512;
  }
}
ge-1/0/9 {
  description to_P1;
  unit 0 {
    family inet {
      address 10.30.0.1/30;
    }
    family mpls;
  }
}
lo0 {
  unit 0 {
    description loopback-interface;
    family inet {
      address 10.255.70.31/32;
    }
  }
}
```


Copyright © 2019, Juniper Networks, Inc.

```
user@PE1# show protocols
mpls {
   interface ge-1/0/9.0;
}
bgp {
   group to_PE2 {
      type internal;
      local-address 10.255.70.31;
      family l2vpn {
         signaling;
      }
      neighbor 172.30.14.1;
   }
}
ospf {
   traffic-engineering;
   area 0.0.0.0 {
      interface lo0.0 {
         passive;
      }
      interface ge-1/0/9.0;
   }
}
ldp {
   interface ge-1/0/9.0;
   interface lo0.0;
}
```

```
user@PE1# show routing-options
router-id 10.255.70.31;
autonomous-system 64511;
```

```
user@PE1# show routing-instances
vpls_a {
   instance-type vpls;
   interface ge-1/0/6.512;
   route-distinguisher 64511:1;
   vrf-target target:64511:1;
   protocols {
      vpls {
         no-tunnel-services;
         site 1 {
            site-identifier 1;
            interface ge-1/0/6.512;
         }
      }
   }
}
```

If you are done configuring Device PE1, enter **commit** from configuration mode.

```
user@P1# show interfaces
ge-1/0/3 {
```

Copyright © 2019, Juniper Networks, Inc.

```
    description to_P2;
    unit 0 {
      family inet {
        address 10.40.0.1/30;
      }
      family mpls;
    }
}
ge-1/0/9 {
    description to_PE1;
    unit 0 {
      family inet {
        address 10.30.0.2/30;
      }
      family mpls;
    }
}
lo0 {
    unit 0 {
      description loopback-interface;
      family inet {
        address 192.168.16.1/32;
      }
    }
}
lo0 {
    unit 0 {
      description loopback-interface;
      family inet {
        address 192.168.16.1/32;
      }
    }
}
```

```
user@P1# show protocols
mpls {
    interface ge-1/0/9.0;
    interface ge-1/0/3.0;
}
ospf {
    traffic-engineering;
    area 0.0.0.0 {
      interface ge-1/0/3.0;
      interface ge-1/0/9.0;
      interface lo0.0 {
        passive;
      }
    }
}
ldp {
    interface ge-1/0/3.0;
    interface ge-1/0/9.0;
    interface lo0.0;
}
```


Copyright © 2019, Juniper Networks, Inc.

```
user@P1# show routing-options
router-id 192.168.16.1;
```

If you are done configuring Device P1, enter **commit** from configuration mode.

```
user@P2# show interfaces
ge-2/0/6 {
   description to_P1;
   unit 0 {
      family inet {
         address 10.40.0.2/30;
      }
      family mpls;
   }
}
ge-2/0/8 {
   description to_PE2;
   unit 0 {
      family inet {
         address 10.50.0.1/30;
      }
      family mpls;
   }
}
lo0 {
   unit 0 {
      description loopback-interface;
      family inet {
         address 192.168.13.1/32;
      }
   }
}
```

```
user@P2# show protocols
mpls {
   interface ge-2/0/6.0;
   interface ge-2/0/8.0;
}
ospf {
   traffic-engineering;
   area 0.0.0.0 {
      interface ge-2/0/6.0;
      interface ge-2/0/8.0;
      interface lo0.0 {
         passive;
      }
   }
}
ldp {
   interface ge-2/0/6.0;
   interface ge-2/0/8.0;
   interface lo0.0;
}
```

Copyright © 2019, Juniper Networks, Inc.

```
user@P2# show routing-options
router-id 192.168.13.1;
```

If you are done configuring Device P2, enter **commit** from configuration mode.

```
user@PE2# show interfaces
ge-2/0/8 {
   description to-R1;
   unit 0 {
      family inet {
         address 10.50.0.2/30;
      }
      family mpls;
   }
}
ge-2/1/1 {
   vlan-tagging;
   encapsulation vlan-vpls;
   unit 512 {
      description to_vpls;
      encapsulation vlan-vpls;
      vlan-id 512;
   }
}
lo0 {
   unit 0 {
      description loopback-interface;
      family inet {
         address 172.30.14.1/32;
      }
   }
}
```

```
user@PE2# show protocols
mpls {
   interface ge-2/0/8.0;
}
bgp {
   group to_PE1 {
      type internal;
      local-address 172.30.14.1;
      family l2vpn {
         signaling;
      }
      neighbor 10.255.70.31;
   }
}
ospf {
   traffic-engineering;
   area 0.0.0.0 {
      interface ge-2/0/8.0;
      interface lo0.0 {
         passive;
      }
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
}
ldp {
    interface ge-2/0/8.0;
    interface lo0.0;
}
```

```
user@PE2# show routing-options
router-id 172.30.14.1;
autonomous-system 64511;
```

```
user@PE2# show routing-instances
vpls_a {
    instance-type vpls;
    interface ge-2/1/1.512;
    route-distinguisher 64511:1;
    vrf-target target:64511:1;
    protocols {
        vpls {
            no-tunnel-services;
            site 2 {
                site-identifier 2;
                interface ge-2/1/1.512;
            }
        }
    }
}
```

If you are done configuring Device PE2, enter **commit** from configuration mode.

```
user@CE2# show interfaces
ge-2/0/7 {
    unit 0 {
        description to-Host2;
        family inet {
            filter {
                input ip-v4;
            }
            address 172.16.80.2/30;
        }
    }
}
ge-2/1/2 {
    vlan-tagging;
    unit 512 {
        description to-PE2;
        vlan-id 512;
        family inet {
            address 10.10.1.2/24;
        }
    }
}
lo0 {
```

Copyright © 2019, Juniper Networks, Inc.

```
    unit 1 {
      description loopback-interface;
      family inet {
        address 192.168.0.2/32;
      }
    }
}
user@CE2# show class-of-service
classifiers {
    ieee-802.1 dscp1 {
      forwarding-class expedited-forwarding {
        loss-priority low code-points ef;
      }
      forwarding-class best-effort {
        loss-priority low code-points be;
      }
    }
}
interfaces {
    ge-2/1/2 {
      unit 512 {
        classifiers {
          ieee-802.1 dscp1;
        }
        rewrite-rules {
          ieee-802.1 ieee1-c2;
        }
      }
    }
}
rewrite-rules {
    ieee-802.1 ieee1-c2 {
      forwarding-class expedited-forwarding {
        loss-priority low code-point 010;
      }
      forwarding-class best-effort {
        loss-priority low code-point 000;
      }
    }
}
user@CE2# show protocols
ospf {
    area 0.0.0.0 {
      interface lo0.1 {
        passive;
      }
      interface ge-2/0/7.0 {
        passive;
      }
      interface ge-2/1/2.512;
    }
}
user@CE2# show firewall
family inet {
    filter ip-v4 {
```


Copyright © 2019, Juniper Networks, Inc.

```
    term tcp80 {
      from {
        port 80;
      }
      then dscp ef;
    }
    term 12345 {
      from {
        port 12345;
      }
      then dscp be;
    }
    term accept {
      then accept;
    }
  }
}
```

If you are done configuring Device CE2, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly by verifying that the DSCP aliases are maintained from Device CE1 to Device CE2.

- Clearing the Firewall Counters on page 467
- Sending Traffic into the Network from TCP HTTP Ports 80 and 12345 and Verifying the Results on page 467

Clearing the Firewall Counters

Purpose Confirm that the firewall counters are cleared.

Action On Device CE2, run the **clear firewall all** command to reset the firewall counters to 0.

```
user@CE2> clear firewall all
```

Sending Traffic into the Network from TCP HTTP Ports 80 and 12345 and Verifying the Results

Purpose Send traffic into the network that can be verified at Device CE2.

Action Configure a new firewall on Device CE2 if you want to verify that the traffic that is being transmitted to Device Host2 from Device Host1 still has the correct DSCP aliases. The following commands create and apply the firewall filter that displays the traffic counts for each code point alias:

```
user@CE2# set firewall family inet filter count term be from dscp be
user@CE2# set firewall family inet filter count term be then count be
```

Copyright © 2019, Juniper Networks, Inc.

```
user@CE2# set firewall family inet filter count term ef from dscp ef
user@CE2# set firewall family inet filter count term ef then count ef
user@CE2# set firewall family inet filter count term accept then accept
user@CE2# set interfaces ge-2/0/7 unit 0 family inet filter output count
```

When you are done configuring Device CE2, enter **commit** from configuration mode.

When you are done testing, you can leave the counting filter in place, or remove it.

1. On Device Host1 use a traffic generator to send 20 TCP packets with a source port of 80 into the network.

   The -s flag sets the source port. The -k flag causes the source port to remain steady instead of incrementing. The -c flag sets the number of packets to 20.

   Repeat the task using a source port of 12345.

   ```
   [user@host1]#  hping 172.16.80.1 -s 80 -k -c 20
   [user@host1]#  hping 172.16.80.1 -s 12345 -k -c 20
   ```

2. On Device CE2, display the firewall counters by using the **show firewall** command.

   ```
   user@CE2> show firewall
   ```

   ```
   show firewall

   Filter: __CE2/count
   Counters:
   Name                                                Bytes              Packets
   be                                                    800
   20
   ef                                                    800
   20
   ```

Meaning The code point aliases set by Device CE1 are maintained across the VPLS backbone and appear intact at Device CE2.

Related Documentation

- Example: Configuring and Applying Scheduler Maps on page 269

Copyright © 2019, Juniper Networks, Inc.

Assigning Rewrite Rules on a Per-Customer Basis Using Policy Maps Overview

Traditionally, packet marking (that is, setting rewrite rules) in Junos OS uses the forwarding class and loss priority that have been determined through a behavior aggregate (BA) classifier or multifield classifier. The forwarding class and loss priority are also used to decide queuing behavior. This approach does not allow rewrite rules to be directly assigned for each customer because of the limited number of combinations of forwarding class and loss priority. When a new customer is added, setting rewrite rules using this approach requires changes to the configuration on the core interfaces, which must be avoided as one mistake can affect traffic from all customers.

An alternative packet marking scheme, available starting in Junos OS 16.1, called policy map, enables you to define rewrite rules on a per-customer basis (that is, for each customer). The policy map makes it possible to use any packet field to identify a given flow and specify a rewrite value for that flow.

A policy map is defined at the **[edit class-of-service** policy-map**]** hierarchy level. The policy map can define the following types of packet marking:

- IPv4 Precedence with the following options:
  - **proto-ip** – Mark the packet for IPv4 to IPv4 traffic.
  - **proto-mpls** – Mark the packet for an IPv4 packet entering an MPLS tunnel.
- IPv4 DSCP with the following options:
  - **proto-ip** – Mark the packet for IPv4 to IPv4 traffic.
  - **proto-mpls** – Mark the packet for an IPv4 packet entering an MPLS tunnel.
- IPv6 DSCP with the following options:
  - **proto-ip** – Mark the packet for IPv6 to IPv6 traffic.
  - **proto-mpls** – Mark the packet for an IPv6 packet entering an MPLS tunnel.
- MPLS EXP with the following options:
  - **all-label** – Mark all labels.
  - **outer-label** – Mark only the outer label.
- IEEE 802.1p with the following options:
  - **outer** – Mark only the outer VLAN header.
  - **outer-and-inner** – Mark both the outer and inner VLAN headers.
- IEEE 802.1ad with the following options:
  - **outer** – Mark only the outer VLAN header.
  - **outer-and-inner** – Mark both the outer and inner VLAN headers.

Copyright © 2019, Juniper Networks, Inc.

NOTE: Creating a policy map requires **enhanced-ip**, **enhanced-ethernet**, or **enhanced-mode** to be configured under **[edit chassis network-services]**.

NOTE: Policy maps have the following configuration restrictions:

- When configuring both **proto-ip** and **proto-mpls** options for **inet-precedence**, **dscp**, or **dscp-ipv6**, you must configure both options with the same code point or code point alias.
- You cannot configure **inet-precedence** and **dscp** in the same policy map.
- In case of MPLS SWAP/PUSH operation, only the new labels are marked on all label-switching routers (LSRs), except the penultimate hop case where if it exposes the next label in the stack, then the exposed label is marked. Therefore, with the penultimate hop, the service label is changed.
- You cannot configure **ieee-802.1** and **ieee-802.1ad** in the same policy map.
- You cannot configure both **outer** and **outer-and-inner** options for **ieee-802.1** and **ieee-802.1ad** code points in the same policy map.
- For IEEE 802.1ad with the **outer-and-inner** option, the discard eligibility (DE) bit is marked only for the outer VLAN header. For the inner VLAN header, only the three CoS Bits are marked.

The policy map can be assigned to a customer through a firewall action on an ingress or egress firewall filter (where the match conditions identify the customer). Alternatively, you can also assign a policy map to an ingress interface or a routing instance. You can assign multiple policy maps to a customer, one for each of the customer's traffic flows. Also, a single policy map can be assigned to multiple customers.

A policy map is executed on a packet just before it is queued, so it overrides any other packet- marking scheme that was previously applied to the packet.

Related Documentation

- Configuring Policy Maps to Assign Rewrite Rules on a Per-Customer Basis on page 471

Copyright © 2019, Juniper Networks, Inc.

Configuring Policy Maps to Assign Rewrite Rules on a Per-Customer Basis

Traditionally, packet marking (that is, setting rewrite rules) in Junos OS uses the forwarding class and loss priority that have been determined through a behavior aggregate (BA) classifier or multifield classifier. The forwarding class and loss priority is also used to decide queuing behavior. This approach does not allow rewrite rules to be directly assigned for each customer because of the limited number of combinations of forwarding class and loss priority. When a new customer is added, setting rewrite rules by using this approach requires changes to the configuration on the core interfaces, which must be avoided as one mistake can affect traffic from all customers.

An alternative packet marking scheme, available starting in Junos OS Release 14.2R3, called policy map, enables you to define rewrite rules on a per-customer basis (that is, for each customer). The policy map makes it possible to use any packet field to identify a given flow and specify a rewrite value for that flow.

To configure and apply policy maps, you must have the following:

- MX Series routers containing MPCs
- Junos OS Release 14.2R3 or later

To assign rewrite rules on a per-customer basis:

1. Configure a policy map.

```
[edit class-of-service policy-map policy-map-name]
user@host# set inet-precedence proto-ip code-point [alias | bits];
user@host# set inet-precedence proto-mpls code-point [alias | bits]
user@host# set dscp proto-ip code-point [alias | bits]
user@host# set dscp proto-mpls code-point [alias | bits]
user@host# set dscp-ipv6 proto-ip code-point [alias | bits]
user@host# set dscp-ipv6 proto-mpls code-point [alias | bits]
user@host# set exp all-label code-point [alias | bits]
user@host# set exp outer-label code-point [alias | bits]
user@host# set ieee-802.1 outer code-point [alias | bits]
user@host# set ieee-802.1 outer-and-inner code-point [alias | bits]
user@host# set ieee-802.1ad outer code-point [alias | bits]
user@host# set ieee-802.1ad outer-and-inner code-point [alias | bits]
```

Copyright © 2019, Juniper Networks, Inc.

NOTE: Policy maps have the following configuration restrictions:

- When configuring both **proto-ip** and **proto-mpls** options for **inet-precedence**, **dscp**, or **dscp-ipv6**, you must configure both options with the same code point or code point alias.
- You cannot configure **inet-precedence** and **dscp** in the same policy map.
- You cannot configure **ieee-802.1** and **ieee-802.1ad** in the same policy map.
- You cannot configure both **outer** and **outer-and-inner** options for **ieee-802.1** and **ieee-802.1ad** code points in the same policy map.
- For MPLS POP operation EXP rewrite, if the inner header is also MPLS, only the **exp** value given with the **mpls** option **all-label** will go into effect.

For example:

```
[edit class-of-service]
user@host# set policy-map pm1 dscp proto-ip code-point 111000
user@host# set policy-map pm1 ieee-802.1 outer code-point 001
```

2. Apply the policy map.

   - Apply the policy map an ingress or egress firewall filter.

     ```
     [edit firewall family protocol-family-name filter filter-name]
     user@host# set term term-name from match-conditions
     user@host# set term term-name then policy-map policy-map-name
     ```

     For example:

     ```
     [edit firewall family inet filter f1]
     user@host# set term t1 from address 10.2.2.0/24
     user@host# set term t1 then policy-map pm1
     ```

     NOTE: In this example, every IPv4 packet arriving from IP address 10.2.2.0/24 is assigned a DSCP value of 111000.

   - Alternatively, apply the policy map to a routing instance.

     ```
     [edit class-of-service]
     user@host# set routing-instances routing-instance-name  policy-map
       policy-map-name
     ```

     For example:

     ```
     [edit class-of-service]
     user@host# set routing-instances r1 policy-map p1
     ```


Copyright © 2019, Juniper Networks, Inc.

NOTE: In this example, every IPv4 packet in routing instance r1 is assigned a DSCP value of 111000.

- Alternatively, apply the policy map directly to an *ingress* interface.

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

For example:

```
[edit class-of-service]
user@host# set interfaces xe-4/0/0 unit 0 policy-map p1
```

Related Documentation

- Assigning Rewrite Rules on a Per-Customer Basis Using Policy Maps Overview on page 469

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 11

Altering Class of Service Values in Packets Exiting the Network Using IPv6 DiffServ

- Resources for CoS with DiffServ for IPv6 on page 475
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Terms and Acronyms for CoS with DiffServ for IPv6 on page 476
- Default DSCP Mappings on page 477
- Default Forwarding Classes on page 478
- Juniper Networks Default Forwarding Classes on page 481
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- Configuring a Firewall Filter for an MF Classifier on Customer Interfaces on page 484
- Applying the Firewall Filter to Customer Interfaces on page 485
- Assigning Forwarding Classes to Output Queues on page 485
- Configuring Rewrite Rules on page 486
- DSCP IPv6 Rewrites and Forwarding Class Maps on page 486
- Applying Rewrite Rules to an Interface on page 487
- Configuring RED Drop Profiles on page 488
- Configuring BA Classifiers on page 488
- Applying a BA Classifier to an Interface on page 489
- Configuring a Scheduler on page 489
- Configuring Scheduler Maps on page 490
- Applying a Scheduler Map to an Interface on page 491
- Example: Configuring DiffServ for IPv6 on page 491

Resources for CoS with DiffServ for IPv6

For additional information about CoS using DiffServ for IPv6, see the following:

- RFC 1924, *A Compact Representation of IPv6 Addresses*
- RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*

Copyright © 2019, Juniper Networks, Inc.

- RFC 2475, *An Architecture for Differentiated Services*
- RFC 2640, *Internet Protocol, Version 6 (IPv6) Specification*
- RFC 2983, *Differentiated Service and Tunnels*
- RFC 3260, *New Terminology and Clarifications for DiffServ*
- RFC 3317, *Differentiated Services Quality of Service Policy Information Base*
- RFC 3513, *IP Version 6 Addressing Architecture*

Related Documentation

- System Requirements for CoS with DiffServ for IPv6 on page 476
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483

System Requirements for CoS with DiffServ for IPv6

To implement CoS with DiffServ for IPv6, your system must meet these minimum requirements:

- Junos OS Release 8.2 or later for MX Series routers
- Junos OS Release 6.3 or later for M Series and T Series routers
- Three Juniper Networks M Series, MX Series, or T Series routers
- For M Series routers, Enhanced FPCs capable of supporting DSCPs and, for MF classifiers, Internet Processor II ASICs

Related Documentation

- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- Example: Configuring DiffServ for IPv6 on page 491

Terms and Acronyms for CoS with DiffServ for IPv6

C

class of service — A set of forwarding class parameters that define different treatment for different traffic flows.

classifier — A method of reading a sequence of bits in a packet header or label and determining the packet's forwarding class.

D

Differentiated Services (DiffServ) — A standards-based method of associating CoS parameters with traffic flows and their forwarding classes.

Differentiated Services code point (DSCP) — Values for a 6-bit field defined for IPv4 and IPv6 packet headers that can be used to enforce CoS distinctions in routers.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- System Requirements for CoS with DiffServ for IPv6 on page 476
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- Example: Configuring DiffServ for IPv6 on page 491

Default DSCP Mappings

Table 54 on page 477 shows the mapping of DiffServ service class meanings (aliases) to DSCPs.

Table 54: Default DSCP Mappings

DiffServ Service Class Alias	IPv4 and IPv6 DSCP Mapping
ef	101110
af11	001010
af12	001100
af13	001110
af21	010010
af22	010100
af23	010110
af31	011010
af32	011100
af33	011110
af41	100010
af42	100100
af43	100110
be	000000
cs1	001000
cs2	010000
cs3	011000
cs4	100000

Copyright © 2019, Juniper Networks, Inc.

Table 54: Default DSCP Mappings (continued)

DiffServ Service Class Alias	IPv4 and IPv6 DSCP Mapping
cs5	101000
nc1/cs6	110000
nc2/cs7	111000

None of the aliases are established by DiffServ specifications. The aliases are well-known only through usage. For example, it is widely accepted that the alias for DSCP **101110** is **ef** (expedited forwarding). The 21 well-known DSCPs establish 5 DiffServ service classes:

- **Best-effort (be)**—The router does not apply any special CoS handling to packets with **000000** in the DiffServ field, a backward compatibility feature. There is usually a high probability that these packets will be dropped under congested network conditions.
- **Assured forwarding (af)**—The router offers a high level of assurance that the packets are delivered as long as the packet flow from the customer stays within a certain service profile (the service provider defines the values). The router accepts excess traffic, but applies a random early detection (RED) drop profile to decide if the excess packets should be dropped and not forwarded. Three drop probabilities (low, medium, and high) are defined for this service class.
- **Expedited forwarding (ef)**—The router delivers assured bandwidth, low loss, low delay, and low delay variation (jitter) end-to-end for packets in this service class. Routers accept excess traffic in this class, but in contrast to assured forwarding, out-of-profile expedited-forwarding packets can be forwarded out of sequence or dropped.
- **Conversational services (cs)**—The router delivers assured (usually low) bandwidth with low delay and jitter for packets in this service class. Packets can be dropped, but never delivered out of sequence. Packetized voice is a good example of a conversational service.
- **Network control (nc)**—The router delivers packets in this service class with a low priority (these packets are not delay-sensitive). Typically, these packets represent routing protocol hello or keepalive messages and loss of these packets jeopardizes proper network operation, so delay is preferable to discard.

Related Documentation

- System Requirements for CoS with DiffServ for IPv6 on page 476
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- Example: Configuring DiffServ for IPv6 on page 491

Default Forwarding Classes

Table 55 on page 479 shows the default forwarding class and packet loss priority (PLP) for the well-known DSCPs. It is important to note that although several DSCPs map to

Copyright © 2019, Juniper Networks, Inc.

the **expedited-forwarding** and **assured-forwarding** classes, by default no resources are assigned to these forwarding classes. All of these settings can be changed through configuration.

Table 55: Default Behavior Aggregate Classification

DSCP and DSCP IPv6	Forwarding Class	PLP
ef	expedited-forwarding	low
af11	assured-forwarding	low
af12	assured forwarding	high
af13	assured forwarding	high
af21	best-effort	low
af22	best-effort	low
af23	best-effort	low
af31	best-effort	low
af32	best-effort	low
af33	best-effort	low
af41	best-effort	low
af42	best-effort	low
af43	best-effort	low
be	best-effort	low
cs1	best-effort	low
cs2	best-effort	low
cs3	best-effort	low
cs4	best-effort	low
cs5	best-effort	low
nc1/cs6	network-control	low
nc2/cs7	network control	low
other	best-effort	low

Copyright © 2019, Juniper Networks, Inc.

Table 56 on page 480 shows the router forwarding classes associated with each well-known DSCP code point and the resources assigned to their output queues.

Table 56: Forwarding Classes and Queues Classification

DCSP Alias	DSCP Bits	Forwarding Class	PLP	Queue
ef	101110	expedited-forwarding	low	1
af11	001010	assured-forwarding	low	2
af12	001100	assured-forwarding	high	2
af13	001110	assured-forwarding	high	2
af21	010010	best-effort	low	0
af22	010100	best-effort	low	0
af23	010110	best-effort	low	0
af31	011010	best-effort	low	0
af32	011100	best-effort	low	0
af33	011110	best-effort	low	0
af41	100010	best-effort	low	0
af42	100100	best-effort	low	0
af43	100110	best-effort	low	0
be	000000	best-effort	low	0
cs1	001000	best-effort	low	0
cs2	010000	best-effort	low	0
cs3	011000	best-effort	low	0
cs4	100000	best-effort	low	0
cs5	101000	best-effort	low	0
nc1/cs6	110000	network-control	low	3
nc2/cs7	111000	network-control	low	3
other	—	best-effort	low	0

Copyright © 2019, Juniper Networks, Inc.

Table 57 on page 481 shows the resources assigned to the four forwarding classes in this example.

Table 57: Resources Assigned to Queues

Queue	Forwarding Class	Transmit Rate	Buffer Size	Priority
0	**be** (data)	40%	40%	Low
1	**ef** (financial)	10%	10%	High
2	**af** (audiovisual)	45%	45%	High (with RED)
3	**nc** (network control)	5%	5%	Low

The table shows how the 95 percent of output link transmission rate and buffer size (queue) resources assigned by default to Q0 (best-effort) are distributed to Q1 (expedited forwarding) and Q2 (assured forwarding). The audiovisual traffic consumes more bandwidth than other applications, but the financial information, although critical, is carried in fewer packets. In keeping with DiffServ specifications, a RED drop profile is applied to the assured forwarding class. The financial data has a strict set of traffic parameters that must be respected.

The three DiffServ assured forwarding classes supported (**af11**, **af12**, and **af13**, with low, medium, and high packet drop probability, respectively) are distinguished by using a low PLP and RED drop profile for **af11** and a high PLP and RED for **af12** and **af13**. All of these parameters should be closely monitored initially for performance and adjusted as necessary.

Related Documentation

- System Requirements for CoS with DiffServ for IPv6 on page 476
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- Example: Configuring DiffServ for IPv6 on page 491

Juniper Networks Default Forwarding Classes

Most M Series routers have only four queues built into the hardware. M120, M320, MX Series, and T Series routers can be configured for up to eight queues. If a classifier does not assign a packet to any other queue (for example, for other than well-known DSCPs that have not been added to the classifier), the packet is assigned by default to the class associated with queue 0 (Q0).

Table 58 on page 482 shows the four forwarding classes and queues to which Juniper Networks classifiers assign a packet based on the DSCP values in arriving packet headers.

Copyright © 2019, Juniper Networks, Inc.

Table 58: Default Forwarding Classes

Forwarding Class Name	Queue
best-effort	queue 0
expedited-forwarding	queue 1
assured-forwarding	queue 2
network-control	queue 3

Each forwarding class has an associated scheduler priority. Only two forwarding classes, **best-effort** and **network-control** (Q0 and Q3), are actually referenced in the default scheduler configuration. However, you can manually configure resources for the **expedited-forwarding** and **assured-forwarding** classes (Q1 and Q2).

The default scheduler settings are not visible in the output of the **show class-of-service** command; rather, they are implicit.

Default Scheduler

```
[edit class-of-service]
schedulers {
   network-control {
      transmit-rate percent 5;
      buffer-size percent 5;
      priority low;
      drop-profile-map loss-priority any protocol any;
      drop-profile terminal;
   }
   best-effort {
      transmit-rate percent 95;
      buffer-size percent 95;
      priority low;
      drop-profile-map loss-priority any protocol any;
      drop-profile terminal;
   }
}
drop-profiles {
   terminal {
      fill-level 100 drop-probability 100;
   }
}
```

By default, the **best-effort** forwarding class (Q0) receives 95 percent of the output link bandwidth and buffer space, and the **network-control** forwarding class (Q3) receives 5 percent of the output link bandwidth and buffer space. The default drop profile provides *tail drop*, where the buffer fills and then discards all packets until there is space in the buffer again. There are no schedulers for the **expedited-forwarding** or **assured-forwarding** classes because by default no resources are assigned to Q1 and Q2.

Copyright © 2019, Juniper Networks, Inc.

All **af** classes other than **af1*x*** are mapped to **best-effort**, since RFC 2597 prohibits a node from aggregating classes. In effect, mapping to **best-effort** implies that the node does not support that class.

Related Documentation

- Understanding How Class of Service Manages Congestion and Controls Service Levels in the Network on page 4
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- Example: Configuring DiffServ for IPv6 on page 491

Roadmap for Configuring CoS with IPv6 DiffServ

To configure class of service (CoS) over IPv6, you must:

- Configure a multifield (MF) classifier for IPv6 to detect packets of interest to CoS and assign the packet to the proper forwarding class independently of Differentiated Services Code Point (DSCP). See "Configuring a Firewall Filter for an MF Classifier on Customer Interfaces" on page 484.

  Next, apply the MF classifier to the appropriate interface. See "Applying the Firewall Filter to Customer Interfaces" on page 485.

- Assign the forwarding classes established by the MF classifier to output queues. See "Assigning Forwarding Classes to Output Queues" on page 485.

- Configure rewrite rules to replace DSCPs on packets received from the customer with the values expected by other routers. See "Configuring Rewrite Rules" on page 486.

  Next, apply the rewrite rules to the appropriate interface. See "Applying Rewrite Rules to an Interface" on page 487.

- Configure behavior aggregate (BA) classifiers for IPv6 on network interfaces because the DSCPs have been explicitly rewritten on the edge routers. See "Configuring BA Classifiers" on page 488.

  Next, apply the BA classifier to the appropriate interface. See "Applying a BA Classifier to an Interface" on page 489.

- Configure random early detection (RED) drop profiles to determine the probability of DiffServ assured forwarding packets being discarded under congested conditions. See "Configuring RED Drop Profiles" on page 488.

- Configure schedulers to assign resources, priorities, and drop profiles to output queues. See "Configuring a Scheduler" on page 489.

- Configure a scheduler map to assign a forwarding class to a scheduler. See "Configuring Scheduler Maps" on page 490.

  Next, apply the scheduler map to the appropriate interface. See "Applying a Scheduler Map to an Interface" on page 491.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Configuring a Firewall Filter for an MF Classifier on Customer Interfaces

You configure an MF classifier for IPv6 to detect packets of interest to CoS and assign the packet to the proper forwarding class independently of DSCP. To configure an MF classifier on a customer-facing link, configure a policer for the expedited forwarding traffic and a firewall filter to classify traffic.

```
[edit firewall]
policer ef-FIN-Policer-Profile {
   if-exceeding {
      bandwidth-percent 10;
      burst-size-limit 2k;
   }
   then loss-priority high;
}
family inet6 {
   filter mf-classifier {
      filter-specific;
      term AV {
         from {
            destination-address {
               0:0:FFFF:172.16.79.11;
            }
         }
         then {
            loss-priority low;
            forwarding-class af-AV-class;
         }
      }
      term Finance {
         from {
            destination-address {
               O:0:FFFF:172.16.79.63;
            }
         }
         then {
            policer ef-FIN-Policer-Profile;
            forwarding-class ef-FIN-class;
         }
      }
      term Network-Control {
         from {
            traffic-class 192; # 192 is the 110000 traffic class.
         }
         then {
            forwarding-class nc-CONTROL-class; # This is network control traffic.
         }
      }
      term Data {
```


Copyright © 2019, Juniper Networks, Inc.

```
        then forwarding-class be-DATA-class; # The rest is data.
    }
  }
}
```

Related Documentation

- Applying the Firewall Filter to Customer Interfaces on page 485
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Applying the Firewall Filter to Customer Interfaces

You apply an MF classifier firewall filter for IPv6 to customer interfaces. To apply an MF classifier firewall filter on customer-facing links, apply the classifier as an input filter at the **[edit interfaces]** hierarchy level.

```
[edit interfaces]
so-0/0/1 {
  unit 0 {
    family inet {
      address 192.168.54.1/24;
    }
    family inet6 {
      filter {
        input mf-classifier;
      }
      address 0:0:FFFF:192.168.54.1/120;
    }
  }
}
```

Related Documentation

- Configuring a Firewall Filter for an MF Classifier on Customer Interfaces on page 484
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Assigning Forwarding Classes to Output Queues

You must assign the forwarding classes established by the MF classifier to output queues. To assign a forwarding class to an output queue, include the **forwarding-classes** statement at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
forwarding-classes {
  queue 0 be-DATA-class;
```

Copyright © 2019, Juniper Networks, Inc.

```
    queue 1 ef-FIN-class;
    queue 2 af-AV-class;
    queue 3 nc-CONTROL-class;
}
```

Related Documentation

- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Configuring Rewrite Rules

You configure rewrite rules to replace DSCPs on packets received from the customer with the values expected by other routers. Rewrite rules use the forwarding class information and packet loss priority (PLP) used internally by the router to establish the DSCP on outbound packets. To configure rewrite rules, include the **rewrite-rules** statement at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
rewrite-rules rewrite-IPv6-dscps {
    forwarding-class be-DATA-class {
        loss-priority low code points 000000;
        loss-priority high code points 000001;
    }
    forwarding-class ef-FIN-class {
        loss-priority low code points 101110;
        loss-priority high code points 101111;
    }
    forwarding-class af-AV-class {
        loss-priority low code points 001010;
        loss-priority high code points 001100;
    }
    forwarding-class nc-CONTROL-class {
        loss-priority low code points 110000;
        loss-priority high code points 110001;
    }
}
```

Related Documentation

- Applying Rewrite Rules to an Interface on page 487
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

DSCP IPv6 Rewrites and Forwarding Class Maps

You cannot configure a DSCP IPv6 rewrite rule and output forwarding class map on the same logical interface (unit). These must be used on different logical interfaces. Although

Copyright © 2019, Juniper Networks, Inc.

a warning is issued, there is nothing in the CLI that prevents this configuration. An error message appears when you attempt to commit the configuration.

This example shows the warning and error message that results when the default DSCP IPv6 rewrite rule is configured on logical interface **ge-1/0/4.0** with output forwarding class map **vg1**.

```
[edit class-of-service]
interfaces {
   ge-1/0/4 {
      unit 0 {
         ##
         ## Warning: DSCP-IPv6 rewrite and forwarding class map not allowed on same unit
         ##
         output-forwarding-class-map vg1;
         rewrite-rules {
            dscp-ipv6 default;
         }
      }
   }
}
```

```
user@router# commit
```

```
[edit class-of-service interfaces ge-1/0/4 unit 0 output-forwarding-class-map]
  'output-forwarding-class-map vg1'
    DSCP-IPv6 rewrite and forwarding class map not allowed on same unit
error: commit failed: (statements constraint check failed)
```

Related Documentation

- Applying Forwarding Classes to Interfaces on page 231

Applying Rewrite Rules to an Interface

To apply the configured rewrite rules, include the **rewrite-rules** statement at the **[edit class-of-service interfaces]** hierarchy level.

```
[edit class-of-service interfaces]
so-0/1/1 {
   unit 0 {
      rewrite-rules {
         dscp-ipv6 rewrite-IPv6-dscps;
      }
   }
}
```

Related Documentation

- Configuring Rewrite Rules on page 486
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Copyright © 2019, Juniper Networks, Inc.

Configuring RED Drop Profiles

You configure RED drop profiles to determine the probability of DiffServ assured forwarding packets being discarded under congested conditions. To configure RED drop profiles for assured forwarding without the PLP bit set and with the PLP bit set, include the **drop-profiles** statement at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
drop-profiles {
  af-AV-normal {
    interpolate {
      fill-level [95 100];
      drop-probability [0 100];
    }
  }
  af-AV-with-PLP {
    interpolate {
      fill-level [60 70 80 90 95];
      drop-probability [80 90 95 97 100];
    }
  }
}
```

Assured forwarding traffic with the PLP bit set has a more aggressive drop probability than traffic without the PLP bit set.

Related Documentation

- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Configuring BA Classifiers

You configure BA classifiers for IPv6 on network interfaces because the DSCPs have been explicitly rewritten on the edge routers. To configure a BA classifier for IPv6 DSCPs, include the **dscp-ipv6** statement and give the classifier a name. Then import the default classifier and specify the forwarding class, loss priority, and code points for each established traffic class at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
classifiers {
  dscp-ipv6 IPv6-classifier {
    import default; # Uses the DSCP default map.
    forwarding-class be-DATA-class {
      loss-priority high code-points 000001;
    }
    forwarding-class ef-FIN-class {
      loss-priority high code-points 101111;
    }
    forwarding-class af-AV-class {
```


Copyright © 2019, Juniper Networks, Inc.

```
        loss-priority high code-points 001100;
      }
      forwarding-class nc-CONTROL-class {
        loss-priority high code-points 110001;
      }
    }
}
```

Related Documentation

- Applying a BA Classifier to an Interface on page 489
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Applying a BA Classifier to an Interface

To apply the configured classifier, include the **classifiers** statement at the **[edit class-of-service interfaces]** hierarchy level.

```
[edit class-of-service interfaces]
so-0/1/1 {
  unit 0 {
    classifiers {
      dscp-ipv6 IPv6-classifier;
    }
  }
}
```

Related Documentation

- Configuring BA Classifiers on page 488
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Configuring a Scheduler

You configure schedulers to assign resources, priorities, and drop profiles to output queues. To configure a scheduler, include the **schedulers** statement at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
schedulers {
  be-DATA-scheduler {
    transmit-rate percent 40;
    buffer-size percent 40;
    priority low;
  }
```

Copyright © 2019, Juniper Networks, Inc.

```
    ef-FIN-scheduler {
        transmit-rate percent 10;
        buffer-size percent 10;
        priority high;
    }
    af-AV-scheduler {
        transmit-rate percent 45;
        buffer-size percent 45;
        priority high;
        drop-profile-map loss-priority low protocol any drop-profile af-AV-normal;
        drop-profile-map loss-priority high protocol any drop-profile af-AV-with-PLP;
    }
    nc-CONTROL-scheduler {
        transmit-rate percent 5;
        buffer-size percent 5;
        priority low;
    }
}
```

Related Documentation

- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Configuring Scheduler Maps

You configure a scheduler map to assign a forwarding class to a scheduler. To configure a scheduler map, include the **scheduler-maps** statement and scheduler name at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
scheduler-maps {
    diffserv-cos-map {
        forwarding-class be-DATA-class scheduler be-DATA-scheduler;
        forwarding-class ef-FIN-class scheduler ef-FIN-scheduler;
        forwarding-class af-AV-class scheduler af-AV-scheduler;
        forwarding-class nc-CONTROL-class scheduler nc-CONTROL-scheduler;
    }
}
```

Related Documentation

- Applying a Scheduler Map to an Interface on page 491
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

Copyright © 2019, Juniper Networks, Inc.

Applying a Scheduler Map to an Interface

To apply the configured scheduler map, include the **scheduler-map** statement at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
interfaces {
   so-1/0/1 {
      scheduler-map diffserv-cos-map;
   }
}
```

Related Documentation

- Configuring Scheduler Maps on page 490
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483
- System Requirements for CoS with DiffServ for IPv6 on page 476
- Example: Configuring DiffServ for IPv6 on page 491

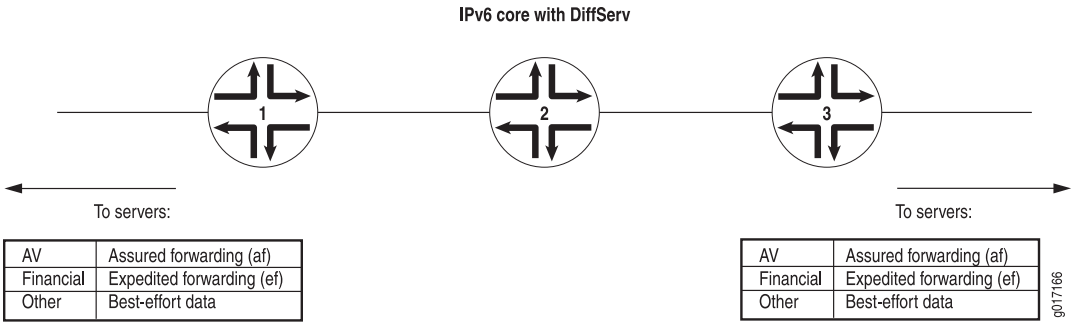
Example: Configuring DiffServ for IPv6

Configuration

The example assigns expedited forwarding to Q1 and a subset of the assured forwarding classes (**af1*x***) to Q2, and distributes resources among all four forwarding classes.

Figure 46 on page 491 shows the topology of the three routers and links that are used as a case study in this chapter.

Figure 46: Basic IPv6 DiffServ Topology



In this case study, the service provider has agreed to provide high-priority delivery of packets for two applications between the customer's servers at two sites. The first application generates streams of high-definition audiovisual (television) packet flows and the second generates large quantities of time-sensitive financial information. In all cases, the packet flow is from server to server. The service provider marks the packets appropriately as they enter the network from either site, configures special queues and forwarding classes for this traffic on the three routers, and uses DiffServ for IPv6 for this purpose.

Copyright © 2019, Juniper Networks, Inc.

Routers 1 and 3 use multifield (MF) classifiers on the customer-facing interfaces to detect high-priority packets and rewrite the Differentiated Services code points (DSCPs) appropriately. Best-effort data and network control packets are not affected. All three routers are configured with consistent schedulers and resources to handle high-priority packets properly.

Figure 47: IPv6 DiffServ Configuration

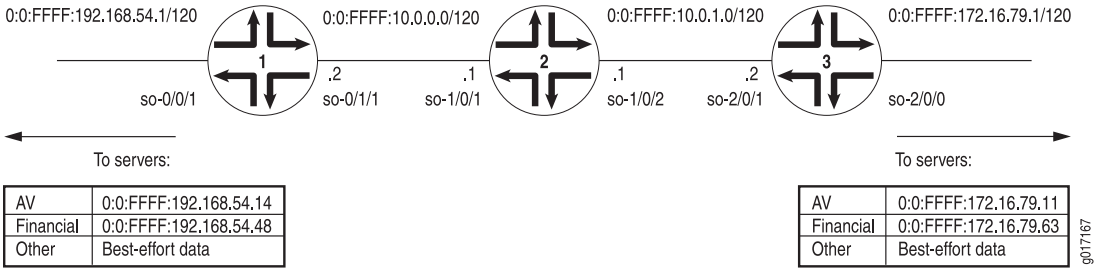


Figure 47 on page 492 shows the complete topology for IPv6 DiffServ, complete with interfaces and IPv6 addresses. The IPv4-mapped IPv6 address format described in RFC 5952 is used.

Begin your configuration on Router 2, the core router. This ensures that when DiffServ is enabled on the edge routers, class of service (CoS) is enabled end to end through the network. The core router configuration is a little simpler because no MF classification is configured in the core.

Router 2

```
[edit]
class-of-service {
  classifiers { # Router 2 classifiers.
    dscp-ipv6 IPv6-classifier {
      import default; # Uses the DSCP default map.
      forwarding-class be-DATA-class {
        loss-priority high code-points 000001;
      }
      forwarding-class ef-FIN-class {
        loss-priority high code-points 101111;
      }
      forwarding-class af-AV-class {
        loss-priority high code-points 001100;
      }
      forwarding-class nc-CONTROL-class {
        loss-priority high code-points 110001;
      }
    }
  }
  drop-profiles { # Router 2 drop profiles.
    af-AV-normal {
      interpolate {
        fill-level [95 100];
        drop-probability [0 100];
      }
    }
    af-AV-with-PLP {
```


Copyright © 2019, Juniper Networks, Inc.

```
      interpolate {
        fill-level [60 70 80 90 95];
        drop-probability [80 90 95 97 100];
      }
    }
  }
  forwarding-classes { # Router 2 forwarding classes.
    queue 0 be-DATA-class;
    queue 1 ef-FIN-class;
    queue 2 af-AV-class;
    queue 3 nc-CONTROL-class;
  }
  interfaces { # Router 2 class-of-service interfaces.
    so-1/0/1 { # Connected to R1.
      scheduler-map diffserv-cos-map;
      unit 0 {
        classifiers {
          dscp-ipv6 IPv6-classifier;
        }
        rewrite-rules {
          dscp-ipv6 rewrite-IPv6-dscp;
        }
      }
    }
    so-1/0/2 { # Connected to R3.
      scheduler-map diffserv-cos-map;
      unit 0 {
        classifiers {
          dscp-ipv6 IPv6-classifier;
        }
        rewrite-rules {
          dscp-ipv6 rewrite-IPv6-dscp;
        }
      }
    }
  }
  rewrite-rules rewrite-IPv6-dscps { # Router 2 rewrite rules.
    forwarding-class be-DATA-class {
      loss-priority low code points 000000;
      loss-priority high code points 000001;
    }
    forwarding-class ef-FIN-class {
      loss-priority low code points 101110;
      loss-priority high code points 101111;
    }
    forwarding-class af-AV-class {
      loss-priority low code points 001010;
      loss-priority high code points 001100;
    }
    forwarding-class nc-CONTROL-class {
      loss-priority low code points 110000;
      loss-priority high code points 110001;
    }
  }
  scheduler-maps { # Router 2 scheduler maps.
```

Copyright © 2019, Juniper Networks, Inc.

```
    diffserv-cos-map {
      forwarding-class be-DATA-class scheduler be-DATA-scheduler;
      forwarding-class ef-FIN-class scheduler ef-FIN-scheduler;
      forwarding-class af-AV-class scheduler af-AV-scheduler;
      forwarding-class nc-CONTROL-class scheduler nc-CONTROL-scheduler;
    }
  }
  schedulers { # Router 2 schedulers.
    be-DATA-scheduler {
      transmit-rate percent 40;
      buffer-size percent 40;
      priority low;
    }
    ef-FIN-scheduler {
      transmit-rate percent 10;
      buffer-size percent 10;
      priority high;
    }
    af-AV-scheduler {
      transmit-rate percent 45;
      buffer-size percent 45;
      priority high;
      drop-profile-map loss-priority low protocol any drop-profile af-AV-normal;
      drop-profile-map loss-priority high protocol any drop-profile af-AV-with-PLP;
    }
    nc-CONTROL-scheduler {
      transmit-rate percent 5;
      buffer-size percent 5;
      priority low;
    }
  }
}
interfaces { # R2 interfaces.
  so-1/0/1 { # Connected to R1.
    unit 0 {
      family inet {
        address 10.0.0.1/24;
      }
      family inet6 {
        address 0:0:FFFF:10.0.0.1/120;
      }
    }
  }
  so-1/0/2 { # Connected to R3.
    unit 0 {
      family inet {
        address 10.0.1.1/24;
      }
      family inet6 {
        address 0:0:FFFF:10.0.1.1/120;
      }
    }
  }
}
```


Copyright © 2019, Juniper Networks, Inc.

Continue your configuration on Router 1 and Router 3, the edge routers. These routers get firewall-filter-based MF classifiers and rewrite rules for markers as well as schedulers and drop profiles on the core-facing interfaces.

Router 1

```
[edit]
class-of-service {
  classifiers { # Router 1 classifiers.
    dscp-ipv6 IPv6-classifier {
      import default; # Uses the DSCP default map.
      forwarding-class be-DATA-class {
        loss-priority high code-points 000001;
      }
      forwarding-class ef-FIN-class {
        loss-priority high code-points 101111;
      }
      forwarding-class af-AV-class {
        loss-priority high code-points 001100;
      }
      forwarding-class nc-CONTROL-class {
        loss-priority high code-points 110001;
      }
    }
  }
  drop-profiles { # Router 1 drop profiles.
    af-AV-normal {
      interpolate {
        fill-level [95 100];
        drop-probability [0 100];
      }
    }
    af-AV-with-PLP {
      interpolate {
        fill-level [60 70 80 90 95];
        drop-probability [80 90 95 97 100];
      }
    }
  }
  forwarding-classes { # Router 1 forwarding classes.
    queue 0 be-DATA-class;
    queue 1 ef-FIN-class;
    queue 2 af-AV-class;
    queue 3 nc-CONTROL-class;
  }
  interfaces { # Router 1 class-of-service interfaces.
    so-0/1/1 { # To servers.
      scheduler-map diffserv-cos-map;
      unit 0 {
        classifiers {
          dscp-ipv6 IPv6-classifier;
        }
        rewrite-rules {
          dscp-ipv6 rewrite-IPv6-dscp;
        }
      }
```

Copyright © 2019, Juniper Networks, Inc.

```
    }
    rewrite-rules rewrite-IPv6-dscps { # Router 1 rewrite rules.
      forwarding-class be-DATA-class {
        loss-priority low code points 000000;
        loss-priority high code points 000001;
      }
      forwarding-class ef-FIN-class {
        loss-priority low code points 101110;
        loss-priority high code points 101111;
      }
      forwarding-class af-AV-class {
        loss-priority low code points 001010;
        loss-priority high code points 001100;
      }
      forwarding-class nc-CONTROL-class {
        loss-priority low code points 110000;
        loss-priority high code points 110001;
      }
    }
    scheduler-maps { # Router 1 scheduler map.
      diffserv-cos-map {
        forwarding-class be-DATA-class scheduler be-DATA-scheduler;
        forwarding-class ef-FIN-class scheduler ef-FIN-scheduler;
        forwarding-class af-AV-class scheduler af-AV-scheduler;
        forwarding-class nc-CONTROL-class scheduler nc-CONTROL-scheduler;
      }
    }
    schedulers { # Router 1 schedulers.
      be-DATA-scheduler {
        transmit-rate percent 40;
        buffer-size percent 40;
        priority low;
      }
      ef-FIN-scheduler {
        transmit-rate percent 10;
        buffer-size percent 10;
        priority high;
      }
      af-AV-scheduler {
        transmit-rate percent 45;
        buffer-size percent 45;
        priority high;
        drop-profile-map loss-priority low protocol any drop-profile af-AV-normal;
        drop-profile-map loss-priority high protocol any drop-profile af-AV-with-PLP;
      }
      nc-CONTROL-scheduler {
        transmit-rate percent 5;
        buffer-size percent 5;
        priority low;
      }
    }
  }
  firewall { # Router 1 firewall policer and filter.
    policer ef-FIN-Policer-Profile {
      if-exceeding {
```


Copyright © 2019, Juniper Networks, Inc.

```
        bandwidth-percent 10;
        burst-size-limit 2k;
      }
      then loss-priority high;
    }
    family inet6 {
      filter mf-classifier {
        filter-specific;
        term AV {
          from {
            destination-address {
              0:0:FFFF:172.16.79.11;
            }
          }
          then {
            loss-priority low;
            forwarding-class af-AV-class;
          }
        }
        term Finance {
          from {
            destination-address {
              O:0:FFFF:172.16.79.63;
            }
          }
          then {
            policer ef-FIN-Policer-Profile;
            forwarding-class ef-FIN-class;
          }
        }
        term Network-Control {
          from {
            traffic-class 192; # 192 is the 110000 traffic class.
          }
          then {
            forwarding-class nc-CONTROL-class; # This is network control traffic.
          }
        }
        term Data {
          then forwarding-class be-DATA-class; # The rest is data.
        }
      }
    }
  }
  interfaces { # Router 1 interfaces.
    so-0/0/1 { # To servers.
      unit 0 {
        family inet {
          address 192.168.54.1/24;
        }
        family inet6 {
          filter {
            input mf-classifier;
          }
          address 0:0:FFFF:192.168.54.1/120;
```

```
        }
      }
    }
    so-0/1/1 { # Connected to R2.
      unit 0 {
        family inet {
          address 10.0.0.2/24;
        }
        family inet6 {
          address 0:0:FFFF:10.0.0.2/120;
        }
      }
    }
  }
}
```

Router 3

```
[edit]
class-of-service {
  classifiers { # Router 3 classifiers.
    dscp-ipv6 IPv6-classifier {
      import default; # Uses the DSCP default map.
      forwarding-class be-DATA-class {
        loss-priority high code-points 000001;
      }
      forwarding-class ef-FIN-class {
        loss-priority high code-points 101111;
      }
      forwarding-class af-AV-class {
        loss-priority high code-points 001100;
      }
      forwarding-class nc-CONTROL-class {
        loss-priority high code-points 110001;
      }
    }
  }
  drop-profiles { # Router 3 drop profiles.
    af-AV-normal {
      interpolate {
        fill-level [95 100];
        drop-probability [0 100];
      }
    }
    af-AV-with-PLP {
      interpolate {
        fill-level [60 70 80 90 95];
        drop-probability [80 90 95 97 100];
      }
    }
  }
  forwarding-classes { # Router 3 forwarding classes.
    queue 0 be-DATA-class;
    queue 1 ef-FIN-class;
    queue 2 af-AV-class;
```


Copyright © 2019, Juniper Networks, Inc.

```
    queue 3 nc-CONTROL-class;
}
interfaces { # Router 3 class-of-service interfaces.
    so-2/0/1 { # To servers.
        scheduler-map diffserv-cos-map;
        unit 0 {
            classifiers {
                dscp-ipv6 IPv6-classifier;
            }
            rewrite-rules {
                dscp-ipv6 rewrite-IPv6-dscp;
            }
        }
    }
    rewrite-rules rewrite-IPv6-dscps { # Router 3 rewrite rules.
        forwarding-class be-DATA-class {
            loss-priority low code points 000000;
            loss-priority high code points 000001;
        }
        forwarding-class ef-FIN-class {
            loss-priority low code points 101110;
            loss-priority high code points 101111;
        }
        forwarding-class af-AV-class {
            loss-priority low code points 001010;
            loss-priority high code points 001100;
        }
        forwarding-class nc-CONTROL-class {
            loss-priority low code points 110000;
            loss-priority high code points 110001;
        }
    }
    scheduler-maps { # Router 3 scheduler map.
        diffserv-cos-map {
            forwarding-class be-DATA-class scheduler be-DATA-scheduler;
            forwarding-class ef-FIN-class scheduler ef-FIN-scheduler;
            forwarding-class af-AV-class scheduler af-AV-scheduler;
            forwarding-class nc-CONTROL-class scheduler nc-CONTROL-scheduler;
        }
    }
    schedulers { # Router 3 schedulers.
        be-DATA-scheduler {
            transmit-rate percent 40;
            buffer-size percent 40;
            priority low;
        }
        ef-FIN-scheduler {
            transmit-rate percent 10;
            buffer-size percent 10;
            priority high;
        }
        af-AV-scheduler {
            transmit-rate percent 45;
            buffer-size percent 45;
            priority high;
```

Copyright © 2019, Juniper Networks, Inc.

```
        drop-profile-map loss-priority low protocol any drop-profile af-AV-normal;
        drop-profile-map loss-priority high protocol any drop-profile af-AV-with-PLP;
    }
    nc-CONTROL-scheduler {
        transmit-rate percent 5;
        buffer-size percent 5;
        priority low;
    }
}
firewall { # Router 3 firewall policer and filter.
    policer ef-FIN-Policer-Profile {
        if-exceeding {
            bandwidth-percent 10;
            burst-size-limit 2k;
        }
        then loss-priority high;
    }
    family inet6 {
        filter mf-classifier {
            filter-specific;
            term AV {
                from {
                    destination-address {
                        0:0:FFFF:172.16.79.11;
                    }
                }
                then {
                    loss-priority low;
                    forwarding-class af-AV-class;
                }
            }
            term Finance {
                from {
                    destination-address {
                        O:0:FFFF:172.16.79.63;
                    }
                }
                then {
                    policer ef-FIN-Policer-Profile;
                    forwarding-class ef-FIN-class;
                }
            }
            term Network-Control {
                from {
                    traffic-class 192; # 192 is the 110000 traffic class.
                }
                then {
                    forwarding-class nc-CONTROL-class; # This is network control traffic.
                }
            }
            term Data {
                then forwarding-class be-DATA-class; # The rest is data.
            }
        }
    }
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
    interfaces { # Router 3 interfaces.
      so-2/0/0 { # To servers.
        unit 0 {
          family inet {
            address 1172.16.79.1/24;
          }
          family inet6 {
            filter {
              input mf-classifier;
            }
            address 0:0:FFFF:172.16.79.1/120;
          }
        }
      }
      so-2/0/1 { # to R2
        unit 0 {
          family inet {
            address 10.0.1.2/24;
          }
          family inet6 {
            address 0:0:FFFF:10.0.1.2/120;
          }
        }
      }
    }
  }
}
```

Verification

To verify that your CoS using IPv6 DiffServ configuration is correct, use the following commands:

- **show class-of-service classifier type dscp-ipv6**
- **show class-of-service rewrite-rule type dscp-ipv6**
- **show class-of-service interface**
- **show class-of-service forwarding-table classifier mapping**
- **show class-of-service forwarding-table rewrite-rule mapping**
- **show class-of-service scheduler-map** ***scheduler-map-name***
- **show class-of-service forwarding-table scheduler-map**

The following section shows the output of these commands used with the configuration example.

DiffServ Classifiers

```
user@R1> show class-of-service classifier type dscp-ipv6
Classifier: dscp-ipv6-default, Code point type: dscp-ipv6, Index: 4
  Code point         Forwarding class                  Loss priority
```

Copyright © 2019, Juniper Networks, Inc.

```
000000              be-DATA-class                   low
000001              be-DATA-class                   low
000010              be-DATA-class                   low
000011              be-DATA-class                   low
000100              be-DATA-class                   low
000101              be-DATA-class                   low
000110              be-DATA-class                   low
000111              be-DATA-class                   low
001000              be-DATA-class                   low
001001              be-DATA-class                   low
001010              af-AV-class                     low
001011              be-DATA-class                   low
001100              af-AV-class                     high
001101              be-DATA-class                   low
001110              af-AV-class                     high
001111              be-DATA-class                   low
010000              be-DATA-class                   low
010001              be-DATA-class                   low
010010              be-DATA-class                   low
010011              be-DATA-class                   low
010100              be-DATA-class                   low
010101              be-DATA-class                   low
010110              be-DATA-class                   low
010111              be-DATA-class                   low
011000              be-DATA-class                   low
011001              be-DATA-class                   low
011010              be-DATA-class                   low
011011              be-DATA-class                   low
011100              be-DATA-class                   low
011101              be-DATA-class                   low
011110              be-DATA-class                   low
011111              be-DATA-class                   low
100000              be-DATA-class                   low
100001              be-DATA-class                   low
100010              be-DATA-class                   low
100011              be-DATA-class                   low
100100              be-DATA-class                   low
100101              be-DATA-class                   low
100110              be-DATA-class                   low
100111              be-DATA-class                   low
101000              be-DATA-class                   low
101001              be-DATA-class                   low
101010              be-DATA-class                   low
101011              be-DATA-class                   low
101100              be-DATA-class                   low
101101              be-DATA-class                   low
101110              ef-FIN-class                    low
101111              be-DATA-class                   low
110000              nc-CONTROL-class                low
110001              be-DATA-class                   low
110010              be-DATA-class                   low
110011              be-DATA-class                   low
110100              be-DATA-class                   low
110101              be-DATA-class                   low
110110              be-DATA-class                   low
110111              be-DATA-class                   low
111000              nc-CONTROL-class                low
111001              be-DATA-class                   low
111010              be-DATA-class                   low
111011              be-DATA-class                   low
```


Copyright © 2019, Juniper Networks, Inc.

```
  111100             be-DATA-class                      low
  111101             be-DATA-class                      low
  111110             be-DATA-class                      low
  111111             be-DATA-class                      low
Classifier: IPv6-classifier, Code point type: dscp-ipv6, Index: 18301
  Code point         Forwarding class                   Loss priority
  000000             be-DATA-class                      low
  000001             be-DATA-class                      high
  000010             be-DATA-class                      low
  000011             be-DATA-class                      low
  000100             be-DATA-class                      low
  000101             be-DATA-class                      low
  000110             be-DATA-class                      low
  000111             be-DATA-class                      low
  001000             be-DATA-class                      low
  001001             be-DATA-class                      low
  001010             af-AV-class                        low
  001011             be-DATA-class                      low
  001100             af-AV-class                        high
  001101             be-DATA-class                      low
  001110             af-AV-class                        high
  001111             be-DATA-class                      low
  010000             be-DATA-class                      low
  010001             be-DATA-class                      low
  010010             be-DATA-class                      low
  010011             be-DATA-class                      low
  010100             be-DATA-class                      low
  010101             be-DATA-class                      low
  010110             be-DATA-class                      low
  010111             be-DATA-class                      low
  011000             be-DATA-class                      low
  011001             be-DATA-class                      low
  011010             be-DATA-class                      low
  011011             be-DATA-class                      low
  011100             be-DATA-class                      low
  011101             be-DATA-class                      low
  011110             be-DATA-class                      low
  011111             be-DATA-class                      low
  100000             be-DATA-class                      low
  100001             be-DATA-class                      low
  100010             be-DATA-class                      low
  100011             be-DATA-class                      low
  100100             be-DATA-class                      low
  100101             be-DATA-class                      low
  100110             be-DATA-class                      low
  100111             be-DATA-class                      low
  101000             be-DATA-class                      low
  101001             be-DATA-class                      low
  101010             be-DATA-class                      low
  101011             be-DATA-class                      low
  101100             be-DATA-class                      low
  101101             be-DATA-class                      low
  101110             ef-FIN-class                       low
  101111             ef-FIN-class                       high
  110000             nc-CONTROL-class                   low
  110001             nc-CONTROL-class                   high
  110010             be-DATA-class                      low
  110011             be-DATA-class                      low
  110100             be-DATA-class                      low
  110101             be-DATA-class                      low
```

Copyright © 2019, Juniper Networks, Inc.

```
  110110            be-DATA-class                    low
  110111            be-DATA-class                    low
  111000            nc-CONTROL-class                 low
  111001            be-DATA-class                    low
  111010            be-DATA-class                    low
  111011            be-DATA-class                    low
  111100            be-DATA-class                    low
  111101            be-DATA-class                    low
  111110            be-DATA-class                    low
  111111            be-DATA-class                    low
```

Rewrite Rules

```
user@R1> show class-of-service rewrite-rule type dscp-ipv6
Rewrite rule: dscp-ipv6-default, Code point type: dscp-ipv6, Index: 20
  Forwarding class                Loss priority       Code point
  be-DATA-class                   low                 000000
  be-DATA-class                   high                000000
  ef-FIN-class                    low                 101110
  ef-FIN-class                    high                101110
  af-AV-class                     low                 001010
  af-AV-class                     high                001100
  nc-CONTROL-class                low                 110000
  nc-CONTROL-class                high                111000
Rewrite rule: rewrite-IPv6-dscp, Code point type: dscp-ipv6, Index: 58077
  Forwarding class                Loss priority       Code point
  be-DATA-class                   low                 000000
  be-DATA-class                   high                000001
  ef-FIN-class                    low                 101110
  ef-FIN-class                    high                101111
  af-AV-class                     low                 001010
  af-AV-class                     high                001100
  nc-CONTROL-class                low                 110000
  nc-CONTROL-class                high                110001
```

Class-of-Service Interfaces

```
user@R1> show class-of-service interface
...
Physical interface: so-0/0/1, Index: 141
Queues supported: 4, Queues in use: 4
  Scheduler map: diffserv-cos-map, Index: -543019056
  Logical interface: so-0/0/1.0, Index: 68
    Object         Name                  Type                      Index
    Rewrite        rewrite-IPv6-dscp     dscp-ipv6                 58077
    Rewrite        exp-default           exp                          21
    Classifier     IPv6-classifier       dscp-ipv6                 18301
    Classifier     exp-default           exp                           5
...
Physical interface: so-0/1/1, Index: 144
Queues supported: 4, Queues in use: 4
  Scheduler map: <default>, Index: -113795564

  Logical interface: so-0/1/1.0, Index: 69
    Object         Name                  Type                      Index
    Rewrite        exp-default           exp                          21
```


Copyright © 2019, Juniper Networks, Inc.

```
   Classifier     exp-default           exp                            5
   Classifier     ipprec-compatibility  ip                             8
```

Classifier Mapping

```
user@R1> show class-of-service forwarding-table classifier mapping
                     Table Index/
Interface      Index      Q num      Table type
so-0/0/1.0        68      18301      IPv6 DSCP
so-0/1/1.0        69          8      IPv4 precedence
```

Rewrite Rule Mapping

```
user@R1> show class-of-service forwarding-table rewrite-rule mapping
Interface      Index   Table index  Type
so-0/1/1.0        68      58077     IPv6 DSCP
```

Scheduler Map

```
user@R1> show class-of-service scheduler-map diffserv-cos-map
Scheduler map: diffserv-cos-map, Index: 1094596010
  Scheduler: be-DATA-scheduler, Forwarding class: be-DATA-class, Index: 14343
    Transmit rate: 40 percent, Rate Limit: none, Buffer size: 40 percent,
    Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP         1    <default-drop-profile>
      Low             TCP             1    <default-drop-profile>
      High            non-TCP         1    <default-drop-profile>
      High            TCP             1    <default-drop-profile>
  Scheduler: ef-FIN-scheduler, Forwarding class: ef-FIN-class, Index: 21707
    Transmit rate: 10 percent, Rate Limit: none, Buffer size: 10 percent,
    Priority: high
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP         1    <default-drop-profile>
      Low             TCP             1    <default-drop-profile>
      High            non-TCP         1    <default-drop-profile>
      High            TCP             1    <default-drop-profile>
  Scheduler: af-AV-scheduler, Forwarding class: af-AV-class, Index: 51704
    Transmit rate: 45 percent, Rate Limit: none, Buffer size: 45 percent,
    Priority: high
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP     61474    af-AV-normal
      Low             TCP         61474    af-AV-normal
      High            non-TCP     65199    af-AV-with-PLP
      High            TCP         65199    af-AV-with-PLP
  Scheduler: nc-CONTROL-scheduler, Forwarding class: nc-CONTROL-class, Index:
50404
    Transmit rate: 5 percent, Rate Limit: none, Buffer size: 5 percent,
    Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP         1    <default-drop-profile>
      Low             TCP             1    <default-drop-profile>
```

Copyright © 2019, Juniper Networks, Inc.

```
     High            non-TCP         1    <default-drop-profile>
     High            TCP             1    <default-drop-profile>
```

user@R1> **show class-of-service forwarding-table scheduler-map**

```
...
Interface: so-0/0/1 (Index: 141, Map index: -543019056, Map type: FINAL,
Num of queues: 4):
  Entry 0 (Scheduler index: 14343, Queue #: 0):
    Tx rate: 0 Kb (40%), Buffer size: 40 percent
Priority low
    PLP high: 1, PLP low: 1, TCP PLP high: 1, TCP PLP low: 1
  Entry 1 (Scheduler index: 21707, Queue #: 1):
    Tx rate: 0 Kb (10%), Buffer size: 10 percent
Priority high
    PLP high: 1, PLP low: 1, TCP PLP high: 1, TCP PLP low: 1
  Entry 2 (Scheduler index: 51704, Queue #: 2):
    Tx rate: 0 Kb (45%), Buffer size: 45 percent
Priority high
    PLP high: 65199, PLP low: 61474, TCP PLP high: 65199, TCP PLP low: 61474
  Entry 3 (Scheduler index: 50404, Queue #: 3):
    Tx rate: 0 Kb (5%), Buffer size: 5 percent
Priority low
    PLP high: 1, PLP low: 1, TCP PLP high: 1, TCP PLP low: 1
...
```

Related Documentation

- System Requirements for CoS with DiffServ for IPv6 on page 476
- Roadmap for Configuring CoS with IPv6 DiffServ on page 483

Copyright © 2019, Juniper Networks, Inc.

PART 3

Configuring Platform-Specific Functionality

- Configuring Class of Service on ACX Series Universal Metro Routers on page 509
- Configuring Class of Service on M Series Multiservice Edge Routers on page 553
- Configuring Class of Service on MX Series 3D Universal Edge Routers on page 567
- Configuring Class of Service on PTX Series Packet Transport Routers on page 599
- Configuring Class of Service on T Series Core Routers on page 649

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 12

Configuring Class of Service on ACX Series Universal Metro Routers

- CoS on ACX Series Routers Features Overview on page 510
- Understanding CoS CLI Configuration Statements on ACX Series Routers on page 511
- Configuring CoS on ACX Series Routers on page 513
- Classifiers and Rewrite Rules at the Global, Physical and Logical Interface Levels Overview on page 514
- Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels on page 515
- Applying DSCP and DSCP IPv6 Classifiers on ACX Series Routers on page 517
- Schedulers Overview for ACX Series Routers on page 517
- Shared and Dedicated Buffer Memory Pools on ACX Series Routers on page 518
- CoS for PPP and MLPPP Interfaces on ACX Series Routers on page 521
- CoS for NAT Services on ACX Series Routers on page 531
- Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic in ACX Series on page 532
- Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic in ACX Series on page 533
- Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface in ACX Series on page 533
- RED Drop Profiles Overview on ACX Series Routers on page 535
- Configuring RED Drop Profiles on ACX Series Routers on page 536
- Hierarchical Class of Service in ACX Series Routers on page 536
- Storm Control on ACX Series Routers Overview on page 550

Copyright © 2019, Juniper Networks, Inc.

CoS on ACX Series Routers Features Overview

NOTE: Unless otherwise noted in the following topics, CoS on ACX Series Universal Metro Routers functions the same as CoS on other routers, and information common CoS functionality can be found in the Configuring Class of Service section of *Class of Service Feature Guide (Routers and EX9200 Switches)*.

The following key CoS features are supported on ACX Series Universal Metro Routers:

- Physical interface-based classifiers at the **[edit class-of-service interfaces *interfaces-name*]** hierarchy level
- Fixed classification for all ingress packets traversing a logical interface to a single forwarding class. Fixed classification is supported on all interface types.
- EXP bits located in each MPLS label and used to encode the CoS value of a packet as it traverses a label-switched path (LSP). To configure global EXP bits, include the **exp** statement at the [**edit class-of-service system-defaults classifiers**] hierarchy level.
- Rewrite rules at the physical and logical interface levels including the following: IP type-of-service (ToS), DSCP, MPLS EXP bit value, and IEEE 802.1p bit value.
- Attachment of the following rewrite rules to the physical interface at the [**edit class-of-service interfaces *interface-name* rewrite-rules**] hierarchy level: IP ToS, DSCP, and IEEE 802.1p bit value.
- Rewrite rules for MPLS EXP bits on the logical interface at the [**edit class-of-service interfaces *interface-name* unit *unit-number* rewrite-rule**] hierarchy level.

NOTE: Fine-grained rewrite is not possible, even when you use multifield filters, because of the application-specific integrated circuit (ASIC) limitation.

Queuing and scheduling features include:

- Support for up to eight forwarding classes.
- Support for up to eight egress queues per port.
- Internal buffer of 2 MB with per-egress queue buffer management.
- Three weighted random early detection (WRED) curves for TCP and one WRED curve for non-TCP. There are two fill levels and two drop probabilities per WRED curve; the drop probability corresponding to the first fill must be zero.
- Strict-priority and weighted deficit round-robin scheduling.
- Multiple strict-priority queues per port.

Copyright © 2019, Juniper Networks, Inc.

- Per-queue committed information rate (CIR) and peak information rate (PIR).
- Per-physical-port shaping.

Queue statistics features include:

- Per-egress-queue enqueue statistics in packets, bytes, packets per second (pps), and bits per second (bps).
- Per-egress-queue transmit statistics in packets, bytes, pps, and bps.
- Per-egress-queue drop statistics in packets and pps.

Related Documentation

- Understanding CoS CLI Configuration Statements on ACX Series Routers on page 511
- Configuring CoS on ACX Series Routers on page 513

Understanding CoS CLI Configuration Statements on ACX Series Routers

ACX Series Universal Metro Routers have some statements or statement options supported on other platforms that are not supported or may not have effect on ACX Series devices.

The following CLI options are not applicable to ACX Series Universal Metro Routers:

```
[edit class-of-service schedulers scheduler-name priority-level]
low;
medium-low;
medium-high;
high;
```

Configure the strict-high-priority queue with unlimited transmission bandwidth so that all traffic receives precedence over any non strict-high priority queues.

At the **[edit class-of-service classifiers *type classifier-name*]** hierarchy level, the **dscp-ipv6** and **ieee-802.1ad** classifier types are not supported. For the **dscp** classifier type, only the outer tag is supported.

The following CLI stanza is not applicable to ACX Series Universal Metro Routers.

```
[edit class-of-service interfaces interface-name]
irb {
  unit logical-unit-number {
    classifiers {
      type (classifier-name | default);
    }
    rewrite-rules {
      dscp (rewrite-name | default);
      dscp-ipv6 (rewrite-name | default);
      exp (rewrite-name | default) protocol protocol-types;
      ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
      inet-precedence (rewrite-name | default);
```

Copyright © 2019, Juniper Networks, Inc.

```
    }
  }
}
```

The following CLI statements are not applicable to ACX Series Universal Metro Routers.

```
[edit class-of-service routing-instances routing-instance-name]
```

```
[edit class-of-service scheduler-map-chassis map-name]
```

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
input-shaping-rate (percent percentage | rate);
input-traffic-control-profile profiler-name shared-instance instance-name;
output-traffic-control-profile profile-name shared-instance instance-name;
per-session-scheduler;
scheduler-map map-name;
shaping-rate rate;
```

```
[edit class-of-service interfaces iinterface-name unit logical-unit-number]
  classifiers {
    type (classifier-name | default);
  }
  rewrite-rules {
  dscp (rewrite-name | default);
  dscp-ipv6 (rewrite-name | default);
  exp (rewrite-name | default) protocol protocol-types;
  exp-push-push-push default;
  exp-swap-push-push default;
  ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
  inet-precedence (rewrite-name | default);
}
```

In the above stanza, **[edit class-of-service *interface-name* unit *logical-unit-number* rewrite-rule exp (rewrite-name | default)]** is supported. However, edit **[class-of-service *interface-name* unit *logical-unit-number* rewrite-rule exp protocol *protocol type*]** is not supported.

```
[edit class-of-service interfaces interface-name interface-set interface-set-name]
excess-bandwidth-share;
internal-node;
output-traffic-control-profile profile-name;
output-traffic-control-profile-remaining profile-name;
```

Related Documentation

- CoS on ACX Series Routers Features Overview on page 510
- Configuring CoS on ACX Series Routers on page 513

Copyright © 2019, Juniper Networks, Inc.

Configuring CoS on ACX Series Routers

Physical interface-based classifiers are supported at the **[edit class-of-service interfaces *interfaces-name*]** hierarchy level. EXP bits are located in each MPLS label and used to encode the CoS value of a packet as it traverses an LSP. To configure global EXP bits, include the **exp** statement at the [**edit class-of-service system-defaults classifiers**] hierarchy level.

To configure CoS on ACX Series routers:

1. Configure the class of service.

```
[edit]
user@host# edit class-of-service
```

2. Configure the rewrite rules.

```
[edit class-of-service]
user@host# edit rewrite-rules (dscp | inet-precedence) rewrite-name
user@host# edit forwarding-class class-name
user@host# set loss-priority low class-name code-points (alias | bits)
```

3. Configure behavior aggregate classifiers for DiffServ CoS.

```
[edit class-of-service]
user@host# edit classifiers (dscp | inet-precedence) classifier-name
user@host# edit forwarding-classes class-name
user@host# set loss-priority class-name code-points (alias | bits)
```

4. Configure expedited forwarding class classifiers.

```
[edit class-of-service classifiers]
user@host# edit forwarding-classes class-name
user@host# set loss-priority class-name code-points (alias | bits)
```

5. Define the forwarding-class mappings.

```
[edit class-of-service]
user@host# edit forwarding-classes class queue-number queue-number
```

6. Configure network control forwarding class classifiers.

```
[edit class-of-service]
user@host# edit forwarding-class class-name
user@host# set loss-priority low class-name code-points (alias | bits)
```

7. Apply the rewrite rules and classifiers to the interfaces.

Copyright © 2019, Juniper Networks, Inc.

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

8. Set the global system default.

```
[edit ]
user@host# edit class-of-service system-defaults classifiers exp classifier-name
```

Related Documentation

- CoS on ACX Series Routers Features Overview on page 510
- Understanding CoS CLI Configuration Statements on ACX Series Routers on page 511

Classifiers and Rewrite Rules at the Global, Physical and Logical Interface Levels Overview

On ACX Series Universal Metro Routers and EX Series switches, CoS supports classification and rewrite at the global level and physical interface levels.

NOTE: The ACX6360 router does not support rewrite rules or Layer 2 (IEEE802.1p and IEEE802.1ad) classifiers.

At a global level, you can define EXP classification.

At a physical interface level, you can define the following features:

- DSCP, DSCP-IPV6, and IPv4 precedence classifiers
- DSCP, DSCP-IPV6, and IPv4 precedence rewrites
- IEEE 802.1p and IEEE 802.1ad classifiers (inner and outer)
- IEEE 802.1p and IEEE 802.1ad rewrites (outer)

The IEEE 802.1ad classifier uses IEEE 802.1p and DEI bits together.

NOTE: You cannot configure both IEEE 802.1p and IEEE 802.1ad classifiers together at the physical interface level.

At a logical interface level, you can define the fixed classification and EXP rewrites.

To configure global EXP classifiers, include the **classfiers exp *classifier-name*** statement at the **[edit class-of-service system-defaults]** hierarchy level.

To configure classifiers or rewrite rules at the physical interface, include either the **classifiers** statement or the **rewrite-rules** statement at the **[edit class-of-service] interfaces *interface-name*]** hierarchy level.

Copyright © 2019, Juniper Networks, Inc.

To configure fixed classifiers at the logical interface, include the **[edit class-of-service interfaces *interface-name* unit *number* forwarding-class** *fc*] or the **rewrite-rules** statement at the **[edit class-of-service interfaces *interface-name*]** hierarchy level.

To configure EXP rewrite at the logical interface, include the **[edit class-of-service interfaces *interface-name* unit *number* rewrite-rules exp** *rewrite-rule*] statement.

To display classifiers configured under **system-defaults**, enter the **show class-of-service system-defaults** command.

To display classifiers and rewrite rules bound to physical interfaces, enter the **show class-of-service interfaces *interface-name*** command.

Related Documentation

- Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels on page 515

Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels

On ACX Series Universal Metro Routers and EX Series switches, CoS supports classification and rewrite at the global and physical interface levels.

To configure the global EXP classifier, include the following statements at the **[edit class-of-service] system-defaults** hierarchy level.

```
[edit class-of-service]
{
  system-defaults
    {
    classifiers exp classifier-name
  }
}
```

CoS supports one global system default classifier of the EXP type, as shown in the following example:

```
[edit class-of-service]
{
  system-defaults {
    classifiers {
      exp exp-classf-core;
    }
  }
}
```

To configure classifiers and rewrite rules at the physical interface level, include the following statements at the **[edit class-of-service] interfaces** hierarchy level.

```
[edit class-of-service]
interfaces {
  interface-name
    classifiers dscp classifier-name
```

Copyright © 2019, Juniper Networks, Inc.

```
    classifiers inet-precedence classifier-name
    classifiers ieee-802.1 [vlan-tag (outer | inner)] classifier-name
    rewrite-rules dscp rewrite-name
    rewrite-rules inet-prec rewrite-name
    rewrite-rules ieee-802.1 rewrite-name
}
```

The following example shows classifiers and rewrite rules configured on physical interfaces:

```
ge-0/1/0 {
  unit 0 {
    rewrite-rules {
      exp custom-exp;
    }
  }
  classifiers {
    dscp d1;
    ieee-802.1 ci;
  }
  rewrite-rules {
    dscp default;
  }
}
  ge-0/1/2 {
    classifiers {
      ieee-802.1 ci;
    }
    rewrite-rules {
      ieee-802.1 ri;
    }
  }
  ge-0/1/3 {
    unit 0 {
      rewrite-rules {
        exp custom-exp2;
      }
    }
  }
  ge-0/1/7 {
    classifiers {
      dscp d1;
    }
  }
  ge-0/1/8 {
    classifiers {
      dscp d1;
    }
  }
```

Related Documentation

- Classifiers and Rewrite Rules at the Global, Physical and Logical Interface Levels Overview on page 514

Copyright © 2019, Juniper Networks, Inc.

Applying DSCP and DSCP IPv6 Classifiers on ACX Series Routers

For ACX Series routers, you cannot apply separate DSCP and DSCP IPv6 classifiers for IPv4 and IPv6 packets on a physical interface. Instead, classifier assignment works as follows:

- If you assign a DSCP classifier only, IPv4 and IPv6 packets are classified using the DSCP classifier.
- If you assign a DSCP IPv6 classifier only, IPv4 and IPv6 packets are classified using the DSCP IPv6 classifier.
- If you assign an IP precedence classifier only, IPv4 and IPv6 packets are classified using the IP precedence classifier. In this case, the lower three bits of the DSCP field are ignored because IPv4 precedence mapping requires the upper three bits only.
- You can assign either DSCP, DSCP IPv6, or IPv4 precedence classifier types on a physical interface.
- You can configure either DSCP, DSCP IPv6, or IPv4 precedence rewrite rules on a physical interface. The rewrite rule applies to both IPv4 and IPv6 packets.
- If you assign either the DSCP or the IP precedence classifier in conjunction with the DSCP IPv6 classifier, the commit fails.

Related Documentation

- Default DSCP and DSCP IPv6 Classifiers on page 41

Schedulers Overview for ACX Series Routers

You use *schedulers* to define the properties 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 random early detection (RED) drop profiles associated with the queue.

You associate the schedulers with forwarding classes by means of *scheduler maps*. You can then associate each scheduler map with an interface, thereby configuring the hardware queues, packet schedulers, and RED processes that operate according to this mapping.

In ACX Series routers, you can configure more than one strict-priority queue per port. The hardware services the queues in the descending order of queue numbers marked as strict priority. All the strict-priority queues are given preferential treatment by the scheduler as long as their shaping rates (or peak information rates) are not met. Unlike MX Series routers, the ACX Series routers configured with queues as *strict-high* at the **[edit class-of-service** schedulers ***scheduler-name*** priority ***strict-high*****]** statement hierarchy, the service is based on queue number and not based on sharing the *strict-high* queues.

Unlike other ACX Series routers, ACX5048 and ACX5096 router supports CIR among strict-priority queues. There is no implicit queue number-based priority among the strict-priority queues. Unlike other ACX Series routers, ACX5048 and ACX5096 router

Copyright © 2019, Juniper Networks, Inc.

supports configuring drop profiles for loss-priority **low**, **medium-high**, and **high** for non-TCP protocols as well.

NOTE: The options **buffer-partition multicast percent** *<0-100>* at the **[edit class-of-service schedulers** *scheduler-name* **buffer-size]** hierarchy level and **multicast** *<0-100>* at the **[edit class-of-service schedulers** *scheduler-name* **shared-buffer-maximum]** hierarchy level are supported only on ACX5048 and ACX5096 routers. For more information, see "Shared and Dedicated Buffer Memory Pools on ACX Series Routers" on page 518.

ACX5448 routers support port-based queueing, scheduling, and shaping. You can configure up to eight queues (virtual output queues) per physical interface (port). Scheduling properties can be applied at both physical as well as logical interface levels. The egress scheduler supports two priority levels (**strict-high** and **low**). Multiple strict-high priority queues and multiple low (default) priority queues can be configured.

By default a port on an ACX5448 router gets a dedicated buffer of 100ms and shared buffer from DRAM. Delay buffer controls the latency of the queue during congestion and maximum number of packets that can be held in a queue. Default buffer size per port is 100ms.

NOTE: In ACX Series routers, the **transmit-rates** statement cannot be configured on strict-priority queues because of hardware limitations.

On an ACX 4000 router, whenever the scheduling and shaping parameters of a port or any of its queues are changed, the entire scheduling configuration on the port is erased and the new configuration is applied. During this window, the traffic pattern does not adhere to user parameters. We recommend configuring scheduling for enabling live traffic.

Related Documentation

- RED Drop Profiles Overview on ACX Series Routers on page 535
- Configuring RED Drop Profiles on ACX Series Routers on page 536
- Shared and Dedicated Buffer Memory Pools on ACX Series Routers on page 518

Shared and Dedicated Buffer Memory Pools on ACX Series Routers

The ACX5048 and ACX5096 router has 12 megabytes of Packet Forwarding Engine (PFE) wide common packet buffer memory that is used to store packets on interface queues. The buffer memory is divided into two pools, shared buffers and dedicated buffers or reserved buffers.

Shared buffers are a global memory pool that the router allocates dynamically to ports as needed, so the buffers are shared among the ports. To configure a maximum amount of shared buffer that the multicast packets can consume, include the **multicast** ***percentage*** CLI statement at the [**edit class-of-service schedulers** ***scheduler-name*** **shared-buffer**

Copyright © 2019, Juniper Networks, Inc.

maximum] hierarchy level. The value that you can specify for **multicast *percentage*** CLI command can be from 0 through 100 percent. If the **multicast *percentage*** CLI statement is not added, then the value defined by the **shared-buffer maximum percent *percentage*** is used for multicast packets as well.

Dedicated buffers or reserved buffers are a memory pool divided equally among the router ports. Each port receives a minimum guaranteed amount of buffer space, dedicated to each port, not shared among ports. To configure a dedicated buffer for multicast packets, include the **buffer-partition multicast *percentage*** CLI statement at the [**edit class-of-service schedulers scheduler-name buffer-size**] hierarchy level. The value that you can specify for **buffer-partition multicast *percentage*** CLI command can be from 0 through 100 percent. If the **buffer-partition multicast *percentage*** CLI statement is not configured, then a default value of 25% is reserved for multicast packets.

NOTE: The total amount of actual queue buffer is defined using the **buffer-size** CLI command.

The ACX5448 router supports delay bandwidth buffer (DBB) for virtual output queues (VOQs). ACX5448 router supports an external DRAM memory, as well as an on-chip buffer (OCB) for storing packet data. A packet is either fully stored in the DRAM or fully stored in the OCB and can consume one or more buffers (upto 40 buffers) depending on the packet size versus the buffer size. A buffer contains a single packet or a part of a single packet.

NOTE: The ACX5448 router does not support buffering for IRB multicast traffic and therefore CLIs for configuring multicast is not supported.

The ACX5448 router does not support **buffer-partition multicast percent** *percentage* option for **buffer-size** and **multicast *percentage*** option for **shared-buffer-maximum**.

To configure shared and dedicated buffers, include the **multicast *percentage*** and **buffer-partition multicast *percentage*** CLI statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
schedulers {
  scheduler-name {
    buffer-size (percent percentage | remainder | temporal microseconds | buffer-partition
      multicast percent percentage );
    shared-buffer maximum (percent percentage | multicast percentage);
  }
}
```

The following is a sample configuration for shared and dedicated buffers in ACX5048 and ACX5096 routers:

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

Copyright © 2019, Juniper Networks, Inc.

```
schedulers schd1{
   buffer-size percent 80;
      buffer-partition {
         multicast {
            percent 30;
         }
      }
      shared-buffer {
         maximum {
            20;
            multicast {
               10;
            }
         }
      }
   }
```

The port gets 50 microseconds worth of reserved buffer. For a 10 Gigabyte port without any shaper, this translates to 62500 Bytes.

In the above sample configuration, the total buffer size allotted for the queue is 80 percent.

Under buffer-partition, the multicast packets get 30 percent of the total buffer-size, which translates to about 24 percent of port-buffer. The unicast packets get the remaining 70 percent of 80 percent of the port buffer, which translates to 56 percent of port buffer.

Under shared-buffer, the multicast packets get up to 10 percent of the total shared buffer. Unicast packets use up to 20 percent of the total shared buffer.

The following is a sample configuration for shared and dedicated buffers in ACX5448 router:

```
[edit class-of-service]
schedulers schd1{
   buffer-size percent 80;
      shared-buffer {
         maximum {
            20;
         }
      }
   }
```

The ACX5448 router has OCB size of 16MB and DRAM size of 6GB. The default buffer size per port is 100 microseconds. The total buffer size for 48X10GE + 4X100GE comes to 11MB. The ACX5448 router supports deep buffering of oversubscribed traffic using external DRAM to queue traffic to oversubscribed ports. The ACX5448 router uses the DRAM-Mix mode by default, which uses DRAM buffers during oversubscription cases. The ACX5448 router supports configuring buffer size (dedicated buffers) per egress queue, which is similar to ACX5000 line of routers.

The ACX6360 routerhas a 39 MB total switch buffer pool. By default, 15 percent of the total buffer pool is allocated to the dedicated buffer pool and the remaining is allocated

 Copyright © 2019, Juniper Networks, Inc.

to the shared buffer pool. If you configure the shared buffer pool as less than 100 percent of the available buffer pool, the remaining buffer space is added to the dedicated buffer pool. You can distribute the shared buffer pool among lossless, lossy, and multicast queues with the following configuration:

```
[edit class-of-service shared-buffer]
user@router# set egress percent 100
user@router# set egress buffer-partition lossless percent percent-value
user@router# set egress buffer-partition lossy percent percent-value
user@router# set egress buffer-partition multicast percent percent-value
```

Related Documentation

- Schedulers Overview for ACX Series Routers on page 517

CoS for PPP and MLPPP Interfaces on ACX Series Routers

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.

CoS functionalities are supported on PPP and MLPPP interfaces. Up to four forwarding classes and four queues are supported per logical interface for PPP and MLPPP packets.

NOTE: CoS for PPP and MLPPP Interfaces is not applicable on ACX5048 and ACX5096 routers.

The Junos OS CoS features provide a set of mechanisms that you can use to provide differentiated services when best-effort traffic delivery is insufficient. In designing CoS applications, you must give careful consideration to your service needs, and you must thoroughly plan and design your CoS configuration to ensure consistency across all routing devices in a CoS domain. You must also consider all the routing devices and other networking equipment in the CoS domain to ensure interoperability among all equipment.

Limitations That are Common for CoS on PPP and MLPPP Interfaces

The following restrictions apply for configuring CoS on PPP and MLPPP interfaces on ACX Series routers:

- For interfaces with PPP encapsulation, you can configure interfaces to support the IPv4, Internet Protocol Control Protocol (IPCP), PPP Challenge Handshake Authentication Protocol (CHAP), and Password Authentication Protocol (PAP) applications.
- Drop timeout, which defines a recovery method for any packets dropped by the member links in a link services or multilink bundle, is not applicable for ACX routers.

Copyright © 2019, Juniper Networks, Inc.

- Loss of traffic occurs during a change of CoS configuration; you cannot modify scheduling attributes instantaneously. The link moves to the down state for PPP, and the protocol is denoted as down for MLPPP interfaces.
- Scheduling and shaping capabilities are based on the CIR-EIR model and are not in accordance with the weighed fair queuing mode. The minimum transmit speed is 32 Kbps, and the minimum difference that can be supported between the transmit rate and shaping rate is also 32 Kbps.
- Buffer size is calculated in terms of packets using 256 bytes as the average packet size. For example, if you configure a 10 percent buffer size for TI interfaces, the buffer allocated as 1.536 Mbps * (10/100) * (0.1 sec) = 15360 bits. The following formula computes the configured queue length:

  Queue length configured = Buffer/average packet size = (15360/256)/8 = 7.5 = 8 packets.

  Because there are no shared buffers, the usage of "buffer-size" and "buffer-size exact" attributes result in the same behavior.
- Only two loss priority levels, namely low and high, are supported. Traffic that arrives from the Packet Forwarding Engine with a medium-high priority is treated as high priority traffic. Although you can configure the medium-high loss priority type when you configure the action for a firewall term, it is considered by the system as high priority traffic.
- A fixed, in-built mapping between forwarding class and queue number as follows is performed: Best-effort is queue 0, expedited-forwarding is queue 1, assured-forwarding is queue 2, and network-control is queue 3
- For WRED configurations, the difference between maximum fill-level and minimum-fill level is a number raised to the power of 2 in terms of number of packets (x^2). Otherwise, the lower fill-level is tuned to turn the difference into a value raised to the power of 2. For example, queues contain a size of 64 packets. If the following configuration is performed:

  ```
  fill-level 50 drop-probability 0;
  fill-level 100 drop-probability 100;
  ```

- For the lower fill level, the minimum number of packets is 32. However, if you specify the fill-level to be 45 instead of 50, the lower fill level is 28. Because 64 - 28, which equals 36 is not a power of 2, the lower fill-level is internally adjusted to convert it to be a number exponentially raised to 2.
- When fragmentation-map is configured, the forwarding-class carrying the multi-class 0 traffic must be assigned the highest priority and the forwarding-class carrying the multi-class 3 traffic must be assigned the lowest priority. Such a configuration is necessary because of the NPU design.

Limitations for CoS on PPP Interfaces

The following restrictions apply for configuring CoS on PPP interfaces on ACX Series routers:

Copyright © 2019, Juniper Networks, Inc.

- The distribution of excess rate between 2 or more queues that contain the same priority occurs on a first-come, first-served basis. For example, consider two Queues configured as follows:

  Q1 : Transmit-rate = 10%, Shaping-rate = 20%

  Q2 : Transmit-rate = 10%, Shaping-rate = 30% on same priority

  The excess rate for Q1 = (20 - 10) = 10%

  The excess rate for Q2 = (30 - 10) = 20%

- The excess rate distribution between Q1 and Q2 does not follow the same ratio but packets in these queues are served in first-come, first-served manner. The shaping rate continues to be honored in such a scenario.

Guidelines for Configuring CoS on PPP and MLPPP Interfaces

Keep the following points in mind when you configure CoS on PPP and MLPPP interfaces:

- You can configure only the any option with the **protocol** statement at the **[edit class-of-service schedulers *scheduler-name* drop-profile-map]** hierarchy level to specify the protocol type for the specified scheduler. You cannot specify the TCP or non-TCP protocol types.
- CoS functionalities for fractional T1 and E1 interfaces are not supported. CoS is supported only for full T1 and E1 interfaces.
- Weighted fair queuing (WFQ) shaping and scheduling model is not supported. Instead of WFQ, CIR-EIR model is supported to handle shaping and scheduling requirements.
- Percentage-based rate configuration is not supported for MLPPP LSQ interfaces; only absolute rate configration in bps is supported.
- Auto-adjustment of shaping and scheduling rates with the addition or deletion of T1/E1 links is not supported. All the limitations applicable for CoS on ACX routers apply for PPP interfaces.
- All the policer limitations on ACX routers for Gigabit Ethernet interfaces are valid for PPP interfaces. This restriction includes ingress and egress policers. Because these limitations are chassis-wide, they are also effective for PPP interfaces.
- All valid configurations specified for MLPPP interfaces with inet address families are also valid for MLPPP interfaces with MPLS address families. For example, EXP classifier as a global classifier is supported for ingress classification and EXP rewrite rule is supported for egress logical interfaces.
- PPP encapsulation is supported on ACX1000, ACX2000, ACX2100, and ACX4000 routers.
- A maximum of 1000 logical interfaces can be supported on an ACX router .
- A maximum of 280 PPP or MLPPP logical interfaces can support drop-profiles on a system. On each MIC, a maximum of 140 PPP or MLPPP interfaces are supported.

Copyright © 2019, Juniper Networks, Inc.

Limitations for CoS on MLPPP Interfaces

The following restrictions apply for configuring CoS on MLPPP interfaces on ACX Series routers:

- Percentage-based configuration for scheduling and shaping parameters is not supported; only absolute rate configuration is supported. As a result, dynamic, swift redjustment of shaping and scheduling settings does not happen with the addition or deletion of T1/E1 links.
- Buffer size is calculated in terms of a single T1 or E1 link speed. Therefore, a temporal value, in microseconds, is used to compute the buffer size for a higher value of the buffer size. For the temporal setting, the queuing algorithm starts dropping packets when it queues more than a computed number of bytes. This maximum is computed by multiplying the transmission rate of the queue by the configured temporal value. The default queue size and percentage-based queue size are not based on the current bandwidth.
- If you configure a scheduler map without a fragmentation map, any scheduler-map configuration including the default settings are applied the same behavior as the exact transmission rate functionality. Priorities of traffic are not honored and no excess rates are provisioned. The forwarding class with no rate configuration receives the minimum fixed rate allocated to it, which is 32 Kbps.
- Support for oversubscription and priority is not available, which might cause inefficient bandwidth utilization. For example, consider a default scheduler map, with the best-effort queue configured with a rate that is equal to 16*T1*(.95) transmit-rate exact.
- The network-control queue is configured with a rate that is equal to 16*T1*(0.05) transmit rate exact. In such a scenario, the following behavior is observed for MLPPP bundle with a single T1 link:

  Traffic that arrives as only best-effort type of traffic is provided with complete bandwidth capacity if no traffic is distributed to any other queue. Traffic that arrives on the only network-control queue is limited to a bandwidth of 1.2288 Mbps, even if no traffic is present on any other queue. When traffic arrives on both the best-effort and network-control queues, an equal division of traffic is done on both the queues because both the queues are within their minimum guarantee rate. Queues other than Best-Effort and Network-Control receive 32 Kbps of exact transmit bandwidth.

  Queues other than best-effort and network-control are assigned 32 Kbps of exact transmit bandwidth.

Consider another example of a default scheduler map, and an MLPPP bundle with two T1 links. In such a scenario, the following behavior is observed for MLPPP bundle with two T1 links:

Traffic that arrives on only the best-effort queue obtains the entire bandwidth capacity if there is no traffic on any other queue. Traffic that arrives on only the network-control queue is limited to 1.2288 Mbps, even when no traffic is passing through any other queue.

Copyright © 2019, Juniper Networks, Inc.

When traffic arrives on both the queues, it is marked at 1.2288 Mbps for the network-control queue and at 1.843200 Mbps for best-effort queue.

For a default scheduler-map with an MLPPP bundle that contains 16 T1 links, the traffic that arrives as only best-effort traffic receives a bandwidth that is equal to (0.95 * 16 * T1) capacity if there is no traffic on any other queue. Traffic that arrives as only network-control traffic is limited to 1.2288 Mbps even if no traffic on any other queue is observed. When traffic arrives on both the queues, it is tagged as 1.2288 Mbps for network-control and (0.95 * 16 * T1) Mbps for best-effort queues. If you configure a scheduler-map with a fragmentation map, two or more classes when configured with same priority receive only the transmit-rate served for them and function as the traffic defined for exact transmit-rate functionality.

Support for oversubscription between two multi-classes on the same priority is not provided. The queue corresponding to which there is no-multiclass entry is moved to the disabled state. Only one-to-one mapping between forwarding-class to multi-class is supported. One forwarding class can send traffic corresponding to only one multi-class.

CoS Functionalities for IPv4 Over PPP Interfaces

The following CoS capabilities are supported on PPP interfaces for IPv4 traffic:

- Ingress Classification can be either specified as fixed classifiers or behavior aggregate (BA) classifiers. Fixed classifiers map all traffic on an interface to the forwarding class. The forwarding class determines the output queue. To configure a fixed classifier, include the **forwarding-class class-name** statement at the **[edit class-of-service interface *interface-name* unit *logical-unit-number*]** hierarchy level.
- BA classification, or CoS value traffic classification, refers to a method of packet classification that uses a CoS configuration to set the forwarding class of a packet based on the CoS value in the IP packet header. The CoS value examined for BA classification purposes can be the Differentiated Services code point (DSCP) value, or the IP precedence value, or EXP bits. The default classifier is based on the IP precedence value.
- To configure the global EXP classifier, include the following statements at the **[edit class-of-service system-defaults]** hierarchy level.

```
[edit class-of-service]
{
  system-defaults
    {
    classifiers exp classifier-name
  }
}
```

CoS supports one global system default classifier of the EXP type, as shown in the following example:

```
[edit class-of-service]
{
  system-defaults {
    classifiers {
```

Copyright © 2019, Juniper Networks, Inc.

```
      exp exp-classf-core;
    }
  }
}
```

- All packets that are received on a logical interface in the ingress direction can be classified to a single forwarding class using fixed classification.
- BA Classification based on the following packet fields is supported at the logical interface level:
  - IPv4 - inet-precedence
  - DSCP

  The following is the configuration stanza for defining BA classifiers:

```
[edit class-of-service]
        classifiers {
            (inet-precedence|dscp ) classifier-name
            {
                forwarding-class class-name loss-priority [low | high]
code-points ([ aliases ] | [ dscp-bits ]);
            }
```

- Queuing and scheduling functionalities comprise the following parameters:
  - Transmit rate per queue
  - Shaping rate per queue
  - WRED
  - Forwarding classes. A maximum of 4 forwarding classes can be defined for PPP interfaces.
- The four priorities supported for logical interface-level queuing are strict-high, medium-high, medium-low, or low. The transmit rate per queue (CIR) is the minimum committed rate can be specified for each queue. The shaping rate per queue (PIR) is the maximum transmit rate can be specified for each queue. For WRED, the default behavior is to enable tail drops. The drop profile configuration enables WRED and enables different drop behavior based on the drop precedence to be entered. Loss priority settings help determine which packets are dropped from the network during periods of congestion. The software supports multiple packet loss priority (PLP) designations: low and high
- Buffer-size can be specific in percentage and temporal configuration. This size is turned into the number of packets per queue, with 256 bytes treated as the average packet size. The following settings can be configured at the queue level:
  - Guaranteed transmit rate (CIR)
  - Shaping rate (PIR)
  - Drop profile
  - Buffer size

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service]
       schedulers scheduler_name {
           transmit-rate (percent number | rate);
           shaping-rate (percent number | rate);
           buffer-size (percent | temporal)
           drop-profile-map map_name;
       }
```

Only 4 forwarding classes and only 4 queues per logical interface are supported. Also, logical interface-based shaping is not supported.

- Packet and byte counters for transmitted and dropped packets are available per queue. These statistical details are displayed using the show interfaces queue command. Aggregation to provide port-level statistics, if needed, is also supported by the system. The logical interface-level statistics are correctly available for egress direction and are displayed in the output, but the statistics pertaining to dropped packets are not considered because of hardware limitations. The following configuration stanza defines the rewrite rules:

```
       [edit class-of-service]
       rewrite-rules {
           (inet-precedence | dscp | exp) rewrite-name
           {
               forwarding-class class-name loss-priority [low | high]
code-point value;
           }
       }
```

Each of the rewrite rules can be attached to an interface by using the following configuration syntax:

All of the firewall features supported on ACX routers are applicable for PPP interfaces for IPv4 packets.

- For rewrite rules, IPv4 or inet precedence and EXP rules are supported. EXP rewrite rules apply only to logical interfaces. You cannot apply EXP rewrite rules to physical interfaces. There are no default EXP rewrite rules. If you want to rewrite the EXP value in MPLS packets, you must configure EXP rewrite rules and apply them to logical interfaces. If no rewrite rules are applied, all MPLS labels that are pushed have a value of zero (0). The EXP value remains unchanged on MPLS labels that are swapped.

CoS Functionalities for IPv4 Over MLPPP Interfaces

The following CoS capabilities are supported on MLPPP interfaces for IPv4 traffic:

- Ingress Classification can either be specified as fixed or BA classifiers.
- Fixed classification using forwarding classes and BA classification using IPv4 precedence value are supported.
- The following scheduling and queuing properties are supported:

Copyright © 2019, Juniper Networks, Inc.

  - Transmit rate per queue
  - Shaping rate per queue
  - WRED
  - Forwarding classes
  - Buffer size per queue
- Forwarding class to multilink class mapping. You use the **multilink-class** statement to map a forwarding class into a multiclass MLPPP (MCML).
- All of the firewall features supported on ACX routers are applicable for MLPPP interfaces for IPv4 packets
- For rewrite rules, IPv4 precedence rule is supported.

The following CoS capabilities are supported on MLPPP interfaces for MPLS packets:

- Ingress Classification can either be specified as fixed or BA classifiers. Fixed classification using forwarding classes and BA classification using EXP bits are supported.
- The following scheduling and queuing properties are supported:
  - Transmit rate per queue
  - Shaping rate per queue
  - WRED
  - Forwarding classes
  - Buffer size per queue
- Forwarding class to multilink class mapping. You use the **multilink-class** statement to map a forwarding class into a multiclass MLPPP (MCML).
- All of the firewall features supported on ACX routers are applicable for MLPPP interfaces for IPv4 packets
- For rewrite rules, the EXP rule is supported.

The following example illustrates an MLPPP CoS configuration set:

```
[edit]
class-of-service {
        classifiers {
            inet-precedence all-traffic-inet {
                forwarding-class assured-forwarding {
                    loss-priority low code-points 101;
                }
                forwarding-class expedited-forwarding {
                    loss-priority low code-points 000;
                }
            }
        }
        drop-profiles {
            plp-low-red {
                fill-level 50 drop-probability 0;
```


Copyright © 2019, Juniper Networks, Inc.

```
                fill-level 100 drop-probability 100;
            }
            plp-high-red {
                fill-level 25 drop-probability 0;
                fill-level 50 drop-probability 100;
            }
        }
        forwarding-classes {
            queue 0 best-effort;
            queue 1 assured-forwarding;
            queue 2 expedited-forwarding;
            queue 3 network-control;
        }

        schedulers {
            evdo-mlppp-best-effort {
                transmit-rate 1M;
                buffer-size percent 80;
                priority medium-high;
            }
            evdo-mlppp-assured-forwarding {
                transmit-rate 500000;
                buffer-size percent 10;
                drop-profile-map loss-priority low protocol any drop-profile
 plp-low-red;
               drop-profile-map loss-priority high protocol any drop-profile
 plp-high-red;
                priority medium-low;
            }
            evdo-mlppp-expedited-forwarding {
                transmit-rate 300000;
                buffer-size percent 5;
                priority low;
            }
            evdo-mlppp-network-control {
                transmit-rate 200000;
                buffer-size percent 5;
                priority strict-high;
            }
        }

        scheduler-maps {
            evdo-mlppp-cos-map {
              forwarding-class best-effort scheduler evdo-mlppp-best-effort;

                forwarding-class assured-forwarding scheduler
evdo-mlppp-assured-forwarding;
                forwarding-class network-control scheduler
evdo-mlppp-network-control;
                forwarding-class expedited-forwarding scheduler
evdo-mlppp-expedited-forwarding;
            }
        }
        fragmentation-maps {
            frag-mlppp {
                forwarding-class {
                    assured-forwarding {
                        multilink-class 2;
                    }
                    expedited-forwarding {
```

```
                        multilink-class 3;
                    }
                    best-effort {
                        multilink-class 1;
                    }
                    network-control {
                        multilink-class 0;
                    }
                }
            }
        }

        interfaces {
            lsq-0/* {
                    unit * {
                        scheduler-map evdo-mlppp-cos-map;
                        fragmentation-map frag-mlppp;
                    }
            }
        }
```

NOTE: In ACX Series routers, the forwarding class and queue mapping is fixed for PPP and MLPPP interfaces.

The following example illustrates an MLPPP firewall configuration set:

```
[edit]
firewall {
            filter classify-evdo-traffic-mlppp {
                interface-specific;
                term signalling {
                    from {
                        dscp ef;
                    }
                    then {
                        count signalling-counter;
     forwarding-class network-control;
     accept;
  }
            }
    term user-speech {
        from {
                    dscp af31;
                }
        then {
            policer user-speech-rate-limit;
                    count user-speech-counter;
                    forwarding-class network-control;
                    accept;
                }
            }
    term ptt-mcs {
        from {
            dscp af11;
        }
        then {
```


Copyright © 2019, Juniper Networks, Inc.

```
            count ptt-mcs-counter;
            forwarding-class network-control;
            accept;
        }
    }
    term user-video {
        from {
            dscp af21;
        }
        then {
            count user-video-counter;
                loss-priority low;
                forwarding-class assured-forwarding;
                accept;
            }
        }
               term user-cmcs {
                   from {
                       dscp af12;
                   }
            then {
                count user-cmcs-counter;
                loss-priority low;
                forwarding-class assured-forwarding;
                accept;
            }
        }
        term ran-rlp-retransmission {
            from {
                dscp af41;
            }
            then {
                count rlp-retransmit-counter;
                loss-priority high;
                forwarding-class assured-forwarding;
                accept;
            }
        }
        term user-best-effort {
            then {
                   count user-be-counter;
         forwarding-class best-effort;
                accept;
            }
        }
    }
}
```

CoS for NAT Services on ACX Series Routers

For packets that go through NAT services, the classification is done when the packet enters the router, and the resulting forwarding class and the packet loss priority are preserved when the packet goes through the inline services interface. When the packet exits the router through a physical interface, the queuing, scheduling, and rewrite rules configured on the egress physical interface are applied on the basis of the forwarding class and packet loss priority derived during ingress.

Copyright © 2019, Juniper Networks, Inc.

NOTE: CoS for NAT services is not applicable on ACX5048 and ACX5096 routers.

Queuing and scheduling are supported on the inline services interface to handle congestion at the inline services interface. Congestion at inline services interface is possible when the interface is oversubscribed. In this case, you would need to selectively drop packets based on the packet's forwarding class and packet loss priority. To drop packets based on the packet's forwarding class and packet loss priority, scheduler map configurations are supported on the inline services interface.

NOTE: In ACX Series routers, you cannot configure classification and rewrite policies on the inline services interface.

Related Documentation

- *Network Address Translation Overview on ACX Series*
- *Network Address Port Translation Overview*
- *Enabling Inline Services Interface on ACX Series*
- *Understanding Service Sets*
- *Service Filters in ACX Series*
- *Guidelines for Applying Service Filters*
- *Service Filter Match Conditions for IPv4 Traffic*
- *Service Filter Match Conditions for IPv4 Traffic*
- *Network Address Translation Address Overload in ACX Series*
- *Configuring Address Pools for Network Address Port Translation (NAPT) Overview*
- *Configuring Service Sets to Be Applied to Services Interfaces*
- *Configuring Queuing and Scheduling on Inline Services Interface*

Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic in ACX Series

This topic provides a summary of the configuration for setting the IEEE 802.1p field in the Ethernet frame header for host outbound traffic (control plane traffic). You can set a global value for the priority code point that applies to all host outbound traffic. Additionally, or alternatively, you can specify that rewrite rules are applied to all host outbound traffic on egress logical interfaces. These are rules that have been previously configured to set the IEEE 802.1p field for data traffic on those interfaces.

To configure the IEEE 802.1p field settings:

1. (Optional) Specify a global default value for the IEEE 802.1p field for all host outbound traffic.

Copyright © 2019, Juniper Networks, Inc.

See "Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic in ACX Series" on page 533.

2. (Optional) Specify that the IEEE 802.1p rewrite rules for the egress logical interfaces are applied to all host outbound traffic on those interfaces.

   See "Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface in ACX Series" on page 533.

Related Documentation

- Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic in ACX Series on page 533
- Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface in ACX Series on page 533

Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic in ACX Series

This topic describes how to configure a global default value for the IEEE 802.1p field for all host outbound traffic on ACX Series routers.

To configure a global default value for the IEEE 802.1p field:

- Specify the value.

```
[edit class-of-service host-outbound-traffic ieee-802.1]
user@host# set default value
```

For example, specify that a value of 010 is applied to all host outbound traffic:

```
[edit class-of-service host-outbound-traffic ieee-802.1]
user@host# set default 010
```

Related Documentation

- Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic in ACX Series on page 532
- Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface in ACX Series on page 533

Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface in ACX Series

This topic describes how to apply rewrite rules for egress logical interfaces to the IEEE 802.1p field for all host outbound traffic on those interfaces on ACX Series routers.

This task requires separately configured rewrite rules that map packet loss priority information to the code point value in the 802.1p field for data traffic on egress logical

Copyright © 2019, Juniper Networks, Inc.

interfaces. See Rewriting Packet Header Information Overview in the *Junos OS Class of Service Configuration Guide*.

To configure the rewrite rules:

1. Configure the CoS rewrite rules to map the forwarding class to the desired value for the 802.1p field.

2. Associate the rewrite rules to the desired egress logical interfaces.

3. (Optional) Configure the forwarding class for host outbound traffic. Do not configure this forwarding class if you want to use the default forwarding class assignment (input classification).

To configure the rewrite rules to apply to the host outbound traffic IEEE 802.1p field:

- Configure the rewrite rules.

```
[edit class-of-service host-outbound-traffic ieee-802.1]
user@host# set rewrite-rules
```

```
[edit class-of-service]
rewrite-rules {
   ieee-802.1 rewrite_foo {
      forwarding-class network-control {
         loss-priority low code-point 101;
      }
   }
}
interfaces {
   ge-1/0/0 {
      unit 100 {
         rewrite-rules {
            ieee-802.1 rewrite_foo vlan-tag outer-and-inner;
         }
      }
   }
}
host-outbound-traffic {
   forwarding-class network-control;
}
host-outbound-traffic {
   ieee-802.1 {
      rewrite-rules;
   }
}
```

Related Documentation

- Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic in ACX Series on page 532

Copyright © 2019, Juniper Networks, Inc.

- Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic in ACX Series on page 533

RED Drop Profiles Overview on ACX Series Routers

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

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

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

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

NOTE: The ACX6360 router does not support drop profiles.

When you configure drop profiles, there are two important values: the queue fullness and the drop probability. The *queue fullness* represents a percentage of the memory used to store packets in relation to the total amount that has been allocated for that specific queue. Similarly, the *drop probability* is a percentage value that correlates to the likelihood that an individual packet is dropped from the network.

The ACX5448 router supports configuring drop profiles (to specify different drop behavior) for **loss-priority** *low*, *medium-high* and *high* for tcp protocol, as well as for non-tcp protocol. Priority setting (either strict-priority or low) for a particular queue does not affect WRED behavior. The default behavior for any queue is to have tail drops. There are two fill levels supported on the ACX5448 router and the first fill level for drop probability should be zero.

NOTE: The ACX5448 router does not support interpolate drop-probability and supports only the discrete method.

You specify drop probabilities at the **[edit class-of-service** drop-profiles ***drop profile-name*** fill-level ***percentage*** drop-probability ***percentage*****;]** class-of-service (COS) configuration hierarchy and reference them in each scheduler configuration. For each scheduler, you can configure four drop profiles as ***high-tcp***, ***medium-high-tcp***, ***low-tcp***, ***non-tcp***. Two fill levels can be specified in each drop-profile map. The drop probability associated with

Copyright © 2019, Juniper Networks, Inc.

the least fill level must be set as zero, otherwise the CLI command will be rejected during commit check.

Related Documentation

- Configuring RED Drop Profiles on ACX Series Routers on page 536
- Schedulers Overview for ACX Series Routers on page 517

Configuring RED Drop Profiles on ACX Series Routers

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

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

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

After you configure a drop profile, you must assign the drop profile to a drop-profile map, and assign the drop-profile map to a scheduler, as discussed in "Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers" on page 359.

Related Documentation

- Schedulers Overview for ACX Series Routers on page 517
- RED Drop Profiles Overview on ACX Series Routers on page 535

Hierarchical Class of Service in ACX Series Routers

ACX5000 Series and ACX 6360 routers support hierarchical class of servicel. Scheduling properties can be applied at both physical as well as logical interface levels. Service providers will be able to support hierarchical class of service at multiple levels to meet the service level agreements and bandwidth allocations for subscribers.

- Hierarchical Scheduling on the Physical Interface on page 537
- Traffic Control Profiles on page 537
- Schedulers on page 538
- Drop Profiles on page 538
- Scheduler Maps on page 538
- Applying the Traffic Control Profiles on page 539
- Subscriber Services on page 539

Copyright © 2019, Juniper Networks, Inc.

Hierarchical Scheduling on the Physical Interface

By default, the queuing mode on all the physical interfaces in the ACX5000 Series and ACX 6360 routers is 8 queues per physical interface (port). In the hierarchical scheduler mode, you can configure up to 3 levels (physical interface, logical interface, and queues) of scheduling.

You can enable hierarchical scheduling by including the **hierarchical-scheduler** CLI command under the interfaces hierarchy as shown below:

```
[edit]
interfaces ge-0/0/1 {
   hierarchical-scheduler;
}
```

NOTE: If you change the physical interface queuing mode from default to hierarchical scheduler mode or vice-versa, the traffic flowing out of the physical interface during the mode change results in a transient loss of traffic data.

NOTE: ACX 6360 routers do not support per-unit scheduling, which enables the configuration of individual output queues per VLAN.

Traffic Control Profiles

The traffic control profiles hold parameters for levels above the queue level of the scheduler hierarchy. The scheduling and shaping configuration on the scheduler nodes are configured using traffic-control-profiles CLI command and scheduler is for queue level. The traffic control profile defines the following characteristics of a scheduler node:

- Scheduler-map
- Shaping rate
- Guaranteed rate

Traffic control profiles can be attached at physical interface and logical interface level. Scheduling and queuing characteristics can be defined for the scheduler node using the shaping-rate and guaranteed-rate. The following is a sample traffic control profile configuration:

```
[edit class-of-service traffic-control-profiles]
tcp-500m-shaping-rate {
   shaping-rate 500m;
}
tcp-ifl0 {
   shaping-rate 200m;
   guaranteed-rate 100m;
```

Copyright © 2019, Juniper Networks, Inc.

```
    scheduler-map tcp-map-ifl0; # Applies scheduler maps to customer VLANs.
}
```

Schedulers

A scheduler defines scheduling and queuing characteristics of a queue and holds the information about the queues, the last level of the hierarchy. The following is a sample scheduler configuration:

```
[edit class-of-service schedulers]
sched-ifl0-q0 {
    priority low;
    transmit-rate 20m;
    buffer-size temporal 100ms;
    drop-profile loss-priority low dp-low;
    drop-profile loss-priority high dp-high;
}
sched-ifl-q1 {
    priority strict-high;
    shaping-rate 20m;
}
```

Drop Profiles

Drop profiles allow you to specify queue specific behavior to drop packets based on WRED profile under congestion. The following is a sample drop profile configuration:

```
[edit class-of-service drop-profiles]
dp-low {
    fill-level 80 drop-probability 0;
    fill-level 100 drop-probability 100;
}
dp-high {
    fill-level 60 drop-probability 0;
    fill-level 80 drop-probability 100;
}
```

Scheduler Maps

A scheduler map is referenced by traffic control profiles to define queues. The scheduler map establishes the number of queues over a scheduler node, associating a forwarding-class with a scheduler. The following is a sample scheduler map configuration:

```
[edit class-of-service scheduler-maps]
sched-map-ifl0 {
    forwarding-class voice scheduler sched-vlan0-qx;
    forwarding-class video scheduler sched-vlan0-qx;
    forwarding-class data scheduler sched-vlan0-qx;
}
tcp-map-vlan1 {
    forwarding-class voice scheduler sched-vlan1-q0;
    forwarding-class video scheduler sched-vlan1-qx;
```


Copyright © 2019, Juniper Networks, Inc.

```
    forwarding-class data scheduler sched-vlan1-qx;
}
tcp-map-vlanx {
    forwarding-class voice scheduler sched-vlanx-qx;
    forwarding-class video scheduler sched-vlanx-qx;
    forwarding-class data scheduler sched-vlanx-qx;
}
```

Applying the Traffic Control Profiles

You can attach the traffic control profiles at various levels of the scheduler hierarchy to achieve hierarchical class of service. The following is a sample configuration to apply traffic control profiles:

NOTE: Although a shaping rate can be applied directly to the physical interface, hierarchical schedulers must use a traffic control profile to hold this parameter.

```
[edit class-of-service interfaces]
ge-1/0/0 {
    output-traffic-control-profile tcp-500m-shaping-rate;
    unit 0 {
        output-traffic-control-profile tcp-vlan0;
    }
}
```

Subscriber Services

ACX5000 line of routers support hierarchical class of service functionality for subscriber services such as Layer 3 VPN, Layer 2 VPN, Ethernet pseudowire (VPWS), and VPLS for logical interface instance on the AC (Access Port).

NOTE: Hierarchical class of service is not supported for Layer 2 bridging (bridge domain VLAN) service.

The following sections explain the hierarchical class of service configuration for subscriber services:

- Configuring hierarchical class of service for Layer 3 VPN Service on page 539
- Configuring hierarchical class of service for Layer 2 VPN (Ethernet Pseudowires) Service on page 541
- Configuring hierarchical class of service for VPLS Service on page 541
- Verifying the hierarchical class of service configurations on page 542

Configuring hierarchical class of service for Layer 3 VPN Service

ACX5000 line of routers can be configured to provide Layer 3 VPN services to subscribers by connecting the UNI port to a CE device. The physical port can be configured to provide

Copyright © 2019, Juniper Networks, Inc.

Layer 3 VPN services to multiple subscribers. You can schedule traffic for different Layer 3 VPN instances based on the SLA parameters agreed with the subscriber.

The following is a sample UNI and NNI logical interface configuration on the PE router providing the Layer 3 VPN service:

```
[edit interfaces]
xe-0/0/1 {
   description “NNI IFL”;
   unit 0 {
      family inet {
         address 100.1.1.1/24;
      }
      family mpls;
   }
}
ge-0/0/1 {
   description “UNI IFL”;
   hierarchical-scheduler;
   flexible-vlan-tagging;
   unit 0 {
      vlan-id 100;
      family inet {
         address 20.20.0.1/24;
      }
   }
   unit 1 {
      vlan-id 101;
      family inet {
         address 20.20.1.1/24;
      }
   }
   unit 2 {
      vlan-id 2;
      family inet {
         address 20.20.2.1/24;
      }
   }
   unit 3 {
      vlan-id 3;
      family inet {
         address 20.20.3.1/24;
      }
   }
   unit 4 {
      vlan-id 4;
      family inet {
         address 20.20.4.1/24;
      }
   }
   ...
}
```

Scheduling can be enabled on the interfaces to achieve hierarchical class of service support for traffic flowing from NNI towards UNI direction.

Copyright © 2019, Juniper Networks, Inc.

Configuring hierarchical class of service for Layer 2 VPN (Ethernet Pseudowires) Service

ACX5000 line of routers can be configured to provide Layer 2 VPN services to subscribers based on Ethernet pseudowires where the UNI port is connected to a CE device. The physical port can be configured to provide Layer 2 VPN services to multiple subscribers. You can schedule traffic for different pseudowires based on the SLA parameters agreed with the subscriber. Hierarchical class of service can be enabled per UNI logical interface represented as the attachment point of the Ethernet pseudowire to achieve the functionality.

The following is a sample to configure the UNI logical interface on the PE router providing the Layer 2 VPN service based on Ethernet pseudowire:

```
[edit interfaces]
ge-0/0/1 {
   hierarchical-scheduler;
   vlan-tagging;
   unit 0 {
      encapsulation vlan-ccc;
      vlan-id 0;
   }
   unit 1 {
      encapsulation vlan-ccc;
      vlan-id 1;
   }
   unit 2 {
      encapsulation vlan-ccc;
      vlan-id 2;
   }
   unit 3 {
      encapsulation vlan-ccc;
      vlan-id 3;
   }
   unit 4 {
      encapsulation vlan-ccc;
      vlan-id 4;
   }
}
```

You can enable scheduling on the interfaces to achieve hierarchical class of service for traffic flowing from NNI towards UNI direction.

Configuring hierarchical class of service for VPLS Service

ACX5000 line of routers can be configured to provide Layer 2 VPN services to subscribers based on VPLS where the UNI port can be connected to a CE device. Subscriber network is attached to UNI logical interface at the PE router and have a VPLS instance. The same physical port can service multiple VPLS instances for various subscribers. The service provider can schedule traffic for different VPLS instances based on the SLA parameters agreed with the subscriber. You can enable hierarchical class of service per UNI logical interface representing the VPLS instance for the subscriber to achieve the functionality.

Copyright © 2019, Juniper Networks, Inc.

The following is a sample to configure the UNI logical interface on the PE router providing the VPLS service:

```
[edit interfaces]
ge-0/0/1 {
  hierarchical-scheduler;
  vlan-tagging;
  encapsulation vlan-vpls;
  unit 0 {
    encapsulation vlan-vpls;
    vlan-id 0;
  }
  unit 1 {
    encapsulation vlan-vpls;
    vlan-id 1;
  }
  unit 2 {
    encapsulation vlan-vpls;
    vlan-id 2;
  }
  unit 3 {
    encapsulation vlan-vpls;
    vlan-id 3;
  }
  unit 4 {
    encapsulation vlan-vpls;
    vlan-id 4;
  }
}
```

Scheduling can be enabled on the interfaces to achieve hierarchical class of service for the traffic flowing from NNI towards UNI direction.

Verifying the hierarchical class of service configurations

You can use the following CLI commands to verify the configuration:

- **show interfaces queue**—Shows physical interface aggregate, physical interface remaining, and logical interface traffic statistics to monitor the traffic received and transmitted. The following are some sample outputs of **show interfaces queue** CLI command:

```
user@host# run show interfaces queue ge-0/0/4.2
  Logical interface ge-0/0/4.2 (Index 555) (SNMP ifIndex 671)
Forwarding classes: 16 supported, 8 in use
Egress queues: 8 supported, 8 in use
Burst size: 0
Queue: 0, Forwarding classes: 8q0
  Queued:
    Packets              :              1121476                  7642 pps
    Bytes                :            587885024              32237416 bps
  Transmitted:
    Packets              :               594964                  3160 pps
    Bytes                :            304621568              12946280 bps
    Tail-dropped packets : Not Available
```


Copyright © 2019, Juniper Networks, Inc.

```
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    Total-dropped packets:                526512                  4482 pps
    Total-dropped bytes  :             283263456              19291136 bps
  Queue Buffer Usage:
    Reserved buffer      :                0 pkts                     0 bytes
    Shared buffer        :                0 pkts                     0 bytes
Queue: 1, Forwarding classes: 8q1
  Queued:
    Packets              :               1121476                  7642 pps
    Bytes                :             587885154              32237416 bps
  Transmitted:
    Packets              :                594959                  3160 pps
    Bytes                :             304619008              12946280 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    Total-dropped packets:                526517                  4482 pps
    Total-dropped bytes  :             283266146              19291136 bps
  Queue Buffer Usage:
    Reserved buffer      :                0 pkts                     0 bytes
    Shared buffer        :                0 pkts                     0 bytes
Queue: 2, Forwarding classes: 8q2
  Queued:
    Packets              :               1119456                  7321 pps
    Bytes                :             595127702              31122568 bps
  Transmitted:
    Packets              :                274601                  1500 pps
    Bytes                :             140595712               6144000 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    Total-dropped packets:                844855                  5821 pps
    Total-dropped bytes  :             454531990              24978568 bps
  Queue Buffer Usage:
    Reserved buffer      :                0 pkts                     0 bytes
    Shared buffer        :                0 pkts                     0 bytes
Queue: 3, Forwarding classes: 8q3
  Queued:
    Packets              :               1119464                  7303 pps
    Bytes                :             595131980              31122568 bps
  Transmitted:
    Packets              :                274602                  1500 pps
    Bytes                :             140596224               6144000 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    Total-dropped packets:                844862                  5803 pps
    Total-dropped bytes  :             454535756              24978568 bps
  Queue Buffer Usage:
    Reserved buffer      :                0 pkts                     0 bytes
    Shared buffer        :                0 pkts                     0 bytes
Queue: 4, Forwarding classes: 8q4
  Queued:
    Packets              :               1121476                  7642 pps
    Bytes                :             587885024              32237416 bps
  Transmitted:
    Packets              :                594964                  3160 pps
    Bytes                :             304621568              12946280 bps
    Tail-dropped packets : Not Available
```

Copyright © 2019, Juniper Networks, Inc.

```
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    Total-dropped packets:                526512                  4482 pps
    Total-dropped bytes  :             283263456              19291136 bps
  Queue Buffer Usage:
    Reserved buffer      :                0 pkts                     0 bytes
    Shared buffer        :                0 pkts                     0 bytes
Queue: 5, Forwarding classes: 8q5
  Queued:
    Packets              :               1121476                  7660 pps
    Bytes                :             587885024              32310560 bps
  Transmitted:
    Packets              :                594964                  3178 pps
    Bytes                :             304621568              13019424 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    Total-dropped packets:                526512                  4482 pps
    Total-dropped bytes  :             283263456              19291136 bps
  Queue Buffer Usage:
    Reserved buffer      :                0 pkts                     0 bytes
    Shared buffer        :                0 pkts                     0 bytes
Queue: 6, Forwarding classes: 8q6
  Queued:
    Packets              :               1121476                  7017 pps
    Bytes                :             587190842              29535136 bps
  Transmitted:
    Packets              :                621684                  3589 pps
    Bytes                :             318302208              14701712 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    Total-dropped packets:                499792                  3428 pps
    Total-dropped bytes  :             268888634              14833424 bps
  Queue Buffer Usage:
    Reserved buffer      :                0 pkts                     0 bytes
    Shared buffer        :                0 pkts                     0 bytes
Queue: 7, Forwarding classes: 8q7
  Queued:
    Packets              :               1121477                  6481 pps
    Bytes                :             586036910              27137704 bps
  Transmitted:
    Packets              :                666066                  3660 pps
    Bytes                :             341025792              14994280 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    Total-dropped packets:                455411                  2821 pps
    Total-dropped bytes  :             245011118              12143424 bps
  Queue Buffer Usage:
    Reserved buffer      :                0 pkts                     0 bytes
    Shared buffer        :                0 pkts                     0 bytes
```

```
user@host# run show interfaces queue ge-0/0/4 aggregate
Physical interface: ge-0/0/4, Enabled, Physical link is Up
  Interface index: 648, SNMP ifIndex: 1763
  Description:  UNI side - connected to ixia 6/9
Forwarding classes: 16 supported, 8 in use
Egress queues: 8 supported, 8 in use
```


Copyright © 2019, Juniper Networks, Inc.

```
Queue: 0, Forwarding classes: 8q0
  Queued:
    Packets              :              16762731               33205 pps
    Bytes                :            8738362264           139058280 bps
  Transmitted:
    Packets              :               8779233               13724 pps
    Bytes                :            4494967296            56220880 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                   0 pps
    RL-dropped bytes     :                     0                   0 bps
    Total-dropped packets:               7983498               19481 pps
    Total-dropped bytes  :            4243394968            82837400 bps
  Queue Buffer Usage:
    Reserved buffer      :               32 pkts                6656 bytes
    Shared buffer        :               52 pkts               10816 bytes
Queue: 1, Forwarding classes: 8q1
  Queued:
    Packets              :              16762732               33237 pps
    Bytes                :            8738359966           139190408 bps
  Transmitted:
    Packets              :               8779363               13756 pps
    Bytes                :            4495033856            56353008 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                   0 pps
    RL-dropped bytes     :                     0                   0 bps
    Total-dropped packets:               7983369               19481 pps
    Total-dropped bytes  :            4243326110            82837400 bps
  Queue Buffer Usage:
    Reserved buffer      :               32 pkts                6656 bytes
    Shared buffer        :               43 pkts                8944 bytes
Queue: 2, Forwarding classes: 8q2
  Queued:
    Packets              :              16754769               30221 pps
    Bytes                :            8826383526           127522040 bps
  Transmitted:
    Packets              :               4052168                6369 pps
    Bytes                :            2074710016            26095472 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                   0 pps
    RL-dropped bytes     :                     0                   0 bps
    Total-dropped packets:              12702601               23852 pps
    Total-dropped bytes  :            6751673510           101426568 bps
  Queue Buffer Usage:
    Reserved buffer      :               32 pkts                6656 bytes
    Shared buffer        :            24232 pkts             5040256 bytes
Queue: 3, Forwarding classes: 8q3
  Queued:
    Packets              :              16754937               30328 pps
    Bytes                :            8826406908           127965968 bps
  Transmitted:
    Packets              :               4052173                6378 pps
    Bytes                :            2074646336            26134456 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                   0 pps
    RL-dropped bytes     :                     0                   0 bps
    Total-dropped packets:              12702764               23950 pps
    Total-dropped bytes  :            6751760572           101831512 bps
  Queue Buffer Usage:
    Reserved buffer      :               32 pkts                6656 bytes
    Shared buffer        :            24205 pkts             5034640 bytes
```

```
Queue: 4, Forwarding classes: 8q4
  Queued:
    Packets              :              16762735               33406 pps
    Bytes                :            8738360722           139886136 bps
  Transmitted:
    Packets              :               8779404               13828 pps
    Bytes                :            4495054848            56648672 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                   0 pps
    RL-dropped bytes     :                     0                   0 bps
    Total-dropped packets:               7983331               19578 pps
    Total-dropped bytes  :            4243305874            83237464 bps
  Queue Buffer Usage:
    Reserved buffer      :               32 pkts                6656 bytes
    Shared buffer        :               46 pkts                9568 bytes
Queue: 5, Forwarding classes: 8q5
  Queued:
    Packets              :              16762734               33285 pps
    Bytes                :            8738359924           139389112 bps
  Transmitted:
    Packets              :               8779416               13788 pps
    Bytes                :            4495060992            56485136 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                   0 pps
    RL-dropped bytes     :                     0                   0 bps
    Total-dropped packets:               7983318               19497 pps
    Total-dropped bytes  :            4243298932            82903976 bps
  Queue Buffer Usage:
    Reserved buffer      :               32 pkts                6656 bytes
    Shared buffer        :               52 pkts               10816 bytes
Queue: 6, Forwarding classes: 8q6
  Queued:
    Packets              :              16762732               27688 pps
    Bytes                :            8721917186           115931832 bps
  Transmitted:
    Packets              :               9618678               12111 pps
    Bytes                :            4924763136            49614432 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                   0 pps
    RL-dropped bytes     :                     0                   0 bps
    Total-dropped packets:               7144054               15577 pps
    Total-dropped bytes  :            3797154050            66317400 bps
  Queue Buffer Usage:
    Reserved buffer      :               24 pkts                4992 bytes
    Shared buffer        :                0 pkts                   0 bytes
Queue: 7, Forwarding classes: 8q7
  Queued:
    Packets              :              16762733               26045 pps
    Bytes                :            8710804790           108947832 bps
  Transmitted:
    Packets              :              10187546               11805 pps
    Bytes                :            5216023552            48359208 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                   0 pps
    RL-dropped bytes     :                     0                   0 bps
    Total-dropped packets:               6575187               14240 pps
    Total-dropped bytes  :            3494781238            60588624 bps
  Queue Buffer Usage:
    Reserved buffer      :               21 pkts                4368 bytes
    Shared buffer        :                0 pkts                   0 bytes
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# run show interfaces queue ge-0/0/4 remaining-traffic
Physical interface: ge-0/0/4, Enabled, Physical link is Up
  Interface index: 648, SNMP ifIndex: 1763
  Description:  UNI side - connected to ixia 6/9
Forwarding classes: 16 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: 8q0
  Queued:
    Packets              :                77501                 6106 pps
    Bytes                :             41609646              26235344 bps
  Transmitted:
    Packets              :                 3206                  243 pps
    Bytes                :              1641472               999248 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    Total-dropped packets:               74295                 5863 pps
    Total-dropped bytes  :             39968174              25236096 bps
  Queue Buffer Usage:
    Reserved buffer      :              8 pkts                  1664 bytes
    Shared buffer        :           1495 pkts                310960 bytes
Queue: 1, Forwarding classes: 8q1
  Queued:
    Packets              :                77489                 6100 pps
    Bytes                :             41444318              26107008 bps
  Transmitted:
    Packets              :                 9330                  732 pps
    Bytes                :              4776960               3002344 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    Total-dropped packets:               68159                 5368 pps
    Total-dropped bytes  :             36667358              23104664 bps
  Queue Buffer Usage:
    Reserved buffer      :              8 pkts                  1664 bytes
    Shared buffer        :           1435 pkts                298480 bytes
Queue: 2, Forwarding classes: 8q2
  Queued:
    Packets              :                77485                 6099 pps
    Bytes                :             41362188              26054080 bps
  Transmitted:
    Packets              :                12417                  975 pps
    Bytes                :              6357504               3996992 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    Total-dropped packets:               65068                 5124 pps
    Total-dropped bytes  :             35004684              22057088 bps
  Queue Buffer Usage:
    Reserved buffer      :              8 pkts                  1664 bytes
    Shared buffer        :           1507 pkts                313456 bytes
Queue: 3, Forwarding classes: 8q3
  Queued:
    Packets              :                77547                 6114 pps
    Bytes                :             41609314              26271632 bps
  Transmitted:
    Packets              :                 3261                  243 pps
    Bytes                :              1646862               999248 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
```

```
    RL-dropped bytes     :                    0                    0 bps
    Total-dropped packets:                74286                 5871 pps
    Total-dropped bytes  :             39962452             25272384 bps
  Queue Buffer Usage:
    Reserved buffer      :                8 pkts                1664 bytes
    Shared buffer        :             1381 pkts                287248 bytes
Queue: 4, Forwarding classes: 8q4
  Queued:
    Packets              :                77502                 6105 pps
    Bytes                :             41450894             26131200 bps
  Transmitted:
    Packets              :                 9349                  732 pps
    Bytes                :              4786688              3002344 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    Total-dropped packets:                68153                 5373 pps
    Total-dropped bytes  :             36664206             23128856 bps
  Queue Buffer Usage:
    Reserved buffer      :                8 pkts                1664 bytes
    Shared buffer        :             1366 pkts                284128 bytes
Queue: 5, Forwarding classes: 8q5
  Queued:
    Packets              :                77480                 6094 pps
    Bytes                :             41358904             26032304 bps
  Transmitted:
    Packets              :                12444                  975 pps
    Bytes                :              6371328              3996992 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    Total-dropped packets:                65036                 5119 pps
    Total-dropped bytes  :             34987576             22035312 bps
  Queue Buffer Usage:
    Reserved buffer      :                8 pkts                1664 bytes
    Shared buffer        :             1552 pkts                322816 bytes
Queue: 6, Forwarding classes: 8q6
  Queued:
    Packets              :                77970                 6099 pps
    Bytes                :             41151002             25749384 bps
  Transmitted:
    Packets              :                30585                 2440 pps
    Bytes                :             15659520              9997088 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    Total-dropped packets:                47385                 3659 pps
    Total-dropped bytes  :             25491482             15752296 bps
  Queue Buffer Usage:
    Reserved buffer      :                3 pkts                 624 bytes
    Shared buffer        :                0 pkts                   0 bytes
Queue: 7, Forwarding classes: 8q7
  Queued:
    Packets              :                77971                 6099 pps
    Bytes                :             41151540             25749384 bps
  Transmitted:
    Packets              :                30585                 2440 pps
    Bytes                :             15659520              9997088 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
```


Copyright © 2019, Juniper Networks, Inc.

```
   RL-dropped bytes     :                    0                    0 bps
   Total-dropped packets:                47386                 3659 pps
   Total-dropped bytes  :             25492020             15752296 bps
 Queue Buffer Usage:
   Reserved buffer      :                3 pkts                  624 bytes
   Shared buffer        :                0 pkts                    0 bytes
```

- **show class-of-service packet-buffer usage**—Shows the total buffer usage of the system. The following is a sample output of the **show class-of-service packet-buffer usage** CLI command:

```
user@host# run show class-of-service packet-buffer usage
            Egress:
            Total Buffer Bytes     :  10652.89 KB  in use out of 12480.00 KB

            Total Buffer Pkts      :  52445        in use out of 61440

            Dedicated Buffer Bytes :  48.14 KB     in use out of 738.16 KB

            Dedicated Buffer Pkts  :  237          in use out of 3634

            Shared Buffer Bytes    :  10604.75 KB  in use out of 11741.84 KB

            Shared Buffer Pkts     :  52208        in use out of 57806
```

You can use the syslog to view the log messages and error reports.

Related Documentation

Copyright © 2019, Juniper Networks, Inc.

Storm Control on ACX Series Routers Overview

A traffic storm is generated when messages are broadcast on a network and each message prompts a receiving node to respond by broadcasting its own messages on the network. This, in turn, prompts further responses, creating a snowball effect. The LAN is suddenly flooded with packets, creating unnecessary traffic that leads to poor network performance or even a complete loss of network service. Storm control enables the router to monitor traffic levels and to drop broadcast, unknown unicast, and multicast (BUM) packets if they exceed the configured limit.

Storm control is applied on the following traffic types:

- Layer 2 broadcast packets
- Layer 2 multicast packets
- Layer 2 unregistered multicast packets
- Layer 2 registered multicast packets
- Layer 2 unknown unicast packets

Storm control functions slightly differently on ACX Series routers compared to other Juniper Networks routers. On ACX Series routers, storm control is only applicable at the physical interface level. No event will be logged when a traffic storm hits an ACX Series router. Also interfaces will not be bound to any default profile. The default action is to drop the packets exceeding the configured bandwidth.

Storm control configuration is done in two steps. The first step is to create a storm control profile. Use the following configuration to create your storm control profile on an ACX Series router:

```
storm-control-profiles {
   foo {
      all {
         bandwidth [percentage | level] <x>;
         [no-unknown-unicast | no-broadcast | no-multicast | no-registered-multicast |
            no-unregistered-multicast]
      }
   }
}
```

The second step in configuring storm control is to bind the profile to physical interface. The following configuration shows how to bind your storm control profile:

```
[edit interfaces]
   ge-0/0/0 {
   ether-options {
      ethernet-switch-profile {
         storm-control foo;
      }
   }
}
```


Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Controlling Network Access Using Traffic Policing Overview on page 115
- *ACX Series Universal Metro Router Overview*
- *ACX1000 and ACX1100 Routers Hardware and CLI Terminology Mapping*
- *ACX2000 and ACX2100 Routers Hardware and CLI Terminology Mapping*

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 13

Configuring Class of Service on M Series Multiservice Edge Routers

- CoS Features and Limitations on M Series and T Series Routers on page 553
- CoS Features and Limitations on M320 Routers with Enhanced III FPCs on page 560
- Packet Flow on Juniper Networks M Series Multiservice Edge Routers on page 561
- Working Around Multifield Classifier Limitations on M Series Routers on page 564
- Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface on page 565

CoS Features and Limitations on M Series and T Series Routers

Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers, as well as M Series Multiservice Edge Routers with enhanced Flexible PIC Concentrators (FPCs), have more CoS capabilities than M Series routers that use other FPC models. Table 59 on page 554 lists some of these the differences.

To determine whether your M Series router is equipped with an enhanced FPC, issue the **show chassis hardware** command. The presence of an enhanced FPC is designated by the **E-FPC** description in the output.

```
user@host> show chassis hardware
Hardware inventory:
Item             Version  Part number Serial number     Description
Chassis                                31959             M7i
Midplane         REV 02   710-008761   CA0209            M7i Midplane
Power Supply 0   REV 04   740-008537   PD10272           AC Power Supply
Routing Engine   REV 01   740-008846   1000396803        RE-5.0
CFEB             REV 02   750-009492   CA0166            Internet Processor IIv1
FPC 0                                                   E-FPC
  PIC 0          REV 04   750-003163   HJ6416            1x G/E, 1000 BASE-SX
  PIC 1          REV 04   750-003163   HJ6423            1x G/E, 1000 BASE-SX
  PIC 2          REV 04   750-003163   HJ6421            1x G/E, 1000 BASE-SX
  PIC 3          REV 02   750-003163   HJ0425            1x G/E, 1000 BASE-SX
FPC 1                                                   E-FPC
  PIC 2          REV 01   750-009487   HM2275            ASP - Integrated
  PIC 3          REV 01   750-009098   CA0142            2x F/E, 100 BASE-TX
```

Copyright © 2019, Juniper Networks, Inc.

Many operations involving the DSCP bits depend on the router and PIC type. For example, some DSCP classification configurations for MPLS and Internet can only be performed on M120 routers, M320 routers with Enhanced Type III FPCs, and MX Series routers only.

Table 59 on page 554 summarizes CoS features and limitations on M Series and T Series routers.

NOTE: The T4000 router supports the lowest of the scaling numbers for classifiers, rewrite rules, and WRED associated with MX Series and T Series routers.

Table 59: CoS Features and Limitations on M Series and T Series Routers

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
Classifiers					

Copyright © 2019, Juniper Networks, Inc.

Table 59: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
Maximum number per FPC or PIC	1	8	64	64 or 58 total	On IQ2 and IQ2E PICs, the CoS classification and CoS rewrite processes are off-loaded from the FPC to the PIC, so the capabilities and limitations of these types of PICs must be taken into consideration. For M Series router FPCs, the one-classifier limit includes the default IP precedence classifier. If you create a new classifier and apply it to an interface, the new classifier does not override the default classifier for other interfaces on the same FPC. In general, the first classifier associated with a logical interface is used. The default classifier can be replaced only when a single interface is associated with the default classifier. Only 58 user-configurable BA classifiers can be attached to logical interfaces on Type-4 FPCs in T640, T1600, or T4000 routers, because six default classifiers are automatically attached to the interfaces. When interfaces on the FPC come up, three default classifiers are installed in the FPC ASIC table: IPv4 and IPv6, MPLS tagging, and multiservices. Next, three default BA classifiers are installed: DSCP IPv6 (9), and MPLS EXP (10), and IP precedence (13). For user-defined BA classifier types (**dscp**, **dscp-ipv6**, **ieee-802.1p**, **ieee-802.1ad**, **inet-precedence**, and **mpls-exp**), you can attach a maximum of 32 classifiers of the same type (including one default classifier) to a logical interface hosted on a Type-4 FPC in a T640, T1600, or T4000 router. You can attach a maximum of 8 user-configured BA classifiers of the same type to a logical interface hosted on an Enhanced Scaling FPC in a T640, T1600, or T4000 router.
dscp	No	Yes	Yes	Yes	On all routers, you cannot configure IP precedence and DiffServ code point (DSCP) classifiers on a single logical interface, because both apply to IPv4 packets.

Copyright © 2019, Juniper Networks, Inc.

Table 59: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
dscp-ipv6	No	Yes	Yes	Yes	For T Series routers, you can apply separate classifiers for IPv4 and IPv6 packets per logical interface. For M Series router enhanced FPCs, you cannot apply separate classifiers for IPv4 and IPv6 packets. Classifier assignment works as follows: • If you assign a DSCP classifier only, IPv4 and IPv6 packets are classified using the DSCP classifier. • If you assign an IP precedence classifier only, IPv4 and IPv6 packets are classified using the IP precedence classifier. The lower three bits of the DSCP field are ignored because IP precedence mapping requires the upper three bits only. • If you assign either the DSCP or the IP precedence classifier in conjunction with the DSCP IPv6 classifier, the commit fails. • If you assign a DSCP IPv6 classifier only, IPv4 and IPv6 packets are classified using the DSCP IPv6 classifier, but the commit displays a warning message.
ieee-802.1p	No	Yes	Yes	Yes	On M Series router enhanced FPCs and T Series routers, if you associate an IEEE 802.1p classifier with a logical interface, you cannot associate any other classifier with that logical interface. For most PICs, if you apply an IEEE 802.1p classifier to a logical interface, you cannot apply non-IEEE classifiers on other logical interfaces on the same physical interface. This restriction does not apply to Gigabit Ethernet IQ2 PICs.
inet-precedence	Yes	Yes	Yes	Yes	On all routers, you cannot assign IP precedence and DSCP classifiers to a single logical interface, because both apply to IPv4 packets.
mpls-exp	Yes	Yes	Yes	Yes	For M Series router FPCs, only the default MPLS EXP classifier is supported; the default MPLS EXP classifier takes the EXP bits 1 and 2 as the output queue number.
Loss priorities based on the Frame Relay discard eligible (DE) bit	No	No	No	No	–

Copyright © 2019, Juniper Networks, Inc.

Table 59: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
Drop Profiles					
Maximum number per FPC or PIC	2	16	32	32	–
Per queue	No	Yes	Yes	Yes	–
Per loss priority	Yes	Yes	Yes	Yes	–
Per Transmission Control Protocol (TCP) bit	No	Yes	Yes	Yes	–
Policing					
Adaptive shaping for Frame Relay traffic	No	No	No	No	–
Traffic policing	Yes	Yes	Yes	Yes	–
Two-rate tricolor marking (TCM)	No	No	Yes	Yes	Allows you to configure up to four loss priorities. Two-rate TCM is supported on T Series routers with Enhanced III FPCs and the T640 Core Router with Enhanced Scaling FPC4.
Virtual channels	No	No	No	No	–
Queuing					

Copyright © 2019, Juniper Networks, Inc.

Table 59: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
Priority	No	Yes	Yes	Yes	Gigabit Ethernet IQ2 PICs support only one queue in the scheduler map with **medium-high**, **high**, or **strict-high** priority. If more than one queue is configured with **high** or **strict-high** priority, the one that appears first in the configuration is implemented as **strict-high** priority. This queue receives unlimited transmission bandwidth. The remaining queues are implemented as **low** priority, which means they might be starved. On the IQE PIC, you can rate-limit the strict-high and high queues. Without this limiting, traffic that requires low latency (delay) such as voice can block the transmission of medium-priority and low-priority packets. Unless limited, high and strict-high traffic is always sent before lower priority traffic. Support for the **medium-low** and **medium-high** queuing priority mappings varies by FPC type.
Per-queue output statistics	No	Yes	Yes	Yes	Per-queue output statistics are shown in the output of the **show interfaces queue** command.
Rewrite Markers					
Maximum number per FPC or PIC	No maximum	No maximum	64	64	On IQ2 and IQ2E PICs, the CoS classification and CoS rewrite processes are off-loaded from the FPC to the PIC, so the capabilities and limitations of these types of PICs must be taken into consideration.
dscp	No	Yes	Yes	Yes	For M Series router Enhanced FPCs, bits 0 through 5 are rewritten, and bits 6 through 7 are preserved. For M320 and T Series router non-IQ FPCs, bits 0 through 5 are rewritten, and bits 6 through 7 are preserved. For M320 and T Series router FPCs, you must decode the loss priority using the firewall filter before you can use loss priority to select the rewrite CoS value. For M320 and T Series router FPCs, Multiservices and Services PIC link services IQ interfaces (**lsq-**) do not support DSCP rewrite markers.

Copyright © 2019, Juniper Networks, Inc.

Table 59: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
dscp-ipv6	No	Yes	Yes	Yes	For M Series router Enhanced FPCs and M320 and T Series router FPCs, bits 0 through 5 are rewritten, and bits 6 through 7 are preserved. For M320 and T Series routers FPCs, you must decode the loss priority using the firewall filter before you can use loss priority to select the rewrite CoS value. For M320 and T Series router FPCs, Multiservices and Services PIC link services IQ interfaces (**lsq-**) do not support DSCP rewrite markers.
frame-relay-de	No	No	No	No	–
ieee-802.1	No	Yes	Yes	Yes	For M Series router enhanced FPCs and T Series router FPCs, fixed rewrite loss priority determines the value for bit 0; queue number (forwarding class) determines bits 1 and 2. For IQ PICs, you can only configure one IEEE 802.1 rewrite rule on a physical port. All logical ports (units) on that physical port should apply the same IEEE 802.1 rewrite rule. On T Series routers only, when you configure IEEE 802.1p rewrite marking on Gigabit Ethernet IQ, Gigabit Ethernet IQ2, Gigabit Ethernet Enhanced IQ (IQE), and Gigabit Ethernet Enhanced IQ2 (IQ2E) PICs, you cannot configure more than eight forwarding classes. This limitation does not apply to M Series routers. On M Series routers, you can configure up to 16 forwarding classes when you configure IEEE 802.1p rewrite marking on any of these PICs.
inet-precedence	Yes	Yes	Yes	Yes	For M Series router FPCs, bits 0 through 2 are rewritten, and bits 3 through 7 are preserved. For M Series router Enhanced FPCs, bits 0 through 2 are rewritten, bits 3 through 5 are cleared, and bits 6 through 7 are preserved. For M320 and T Series routers FPCs, bits 0 through 2 are rewritten and bits 3 through 7 are preserved. For M320 and T Series router FPCs, you must decode the loss priority using the firewall filter before you can use loss priority to select the rewrite CoS value.

Copyright © 2019, Juniper Networks, Inc.

Table 59: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
mpls-exp	Yes	Yes	Yes	Yes	For M320 and T Series router FPCs, you must decode the loss priority using the firewall filter before you can use loss priority to select the rewrite CoS value. For M Series routers FPCs, fixed rewrite loss priority determines the value for bit 0; queue number (forwarding class) determines bits 1 and 2.

Related Documentation

- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53
- Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 361
- Platform Support for Priority Scheduling on page 329
- CoS Features and Limitations on IQ2 and IQ2E PICs (M Series and T Series) on page 795

CoS Features and Limitations on M320 Routers with Enhanced III FPCs

On Juniper Networks M320 Multiservice Edge Routers, CoS features are supported on the following types of Flexible PIC Concentrators (FPCs):

- FPC2, Enhanced II FPC2, and Enhanced III FPC2—Rated at 16 Gbps full duplex
- FPC3, Enhanced II FPC3, and Enhanced III FPC3—Rated at 20 Gbps full duplex

The Enhanced III FPC2 and FPC3 provide different CoS functionality than the standard and Enhanced II FPC2 and FPC3. You can mix the FPC types in a single M320 router, but CoS processing for packets traveling between the Enhanced II FPCs and Enhanced III FPCs differ from the processing of packets traveling between FPCs of the same type. In cases of mixed FPC types, only the least common denominator of CoS functions is supported.

In particular, the drop priority classification behavior is different for packets traveling between Enhanced II and Enhanced III FPCs in an M320 router chassis. In the Enhanced III FPC, the packet is always classified into one of four packet drop priorities whether the **tri-color** statement is configured or not. However, depending on the presence or absence of the **tri-color** statement, the four colors might have a different meaning to the Enhanced II FPC. For more information about the **tri-color** statement, see "Enabling Tricolor Marking and Limitations of Three-Color Policers" on page 173.

Copyright © 2019, Juniper Networks, Inc.

When packets flow from an Enhanced III FPC to an Enhanced II FPC, the drop priority classification behavior is shown in Table 60 on page 561.

Table 60: Drop Priority Classification for Packet Sent from Enhanced III to Enhanced II FPC on M320 Routers

Enhanced III FPC Drop Priority	Enhanced II FPC Drop Priority (Without Tricolor Marking Enabled)	Enhanced II FPC Drop Priority (with Tricolor Marking Enabled)
low	low	low
medium-low	low	medium-low
medium-high	high	medium-high
high	high	high

When packets flow from an Enhanced II FPC without tricolor marking enabled to an Enhanced III FPC, the drop priority classification behavior is shown in Table 61 on page 561.

Table 61: Drop Priority Classification for Packet Sent from Enhanced II FPC Without Tricolor Marking to Enhanced III FPC on M320 Routers

Enhanced II FPC (Without Tricolor Marking Enabled)	Enhanced III FPC
low	low
high	medium-high

When packets flow from an Enhanced II FPC with tricolor marking enabled to an Enhanced III FPC, the drop priority classification behavior is shown in Table 62 on page 561.

Table 62: Drop Priority Classification for Packet Sent from Enhanced II FPC with Tricolor Marking to Enhanced III FPC on M320 Routers

Enhanced II FPC (With Tricolor Marking Enabled)	Enhanced III FPC
low	low
medium-low	medium-low
medium-high	medium-high
high	high

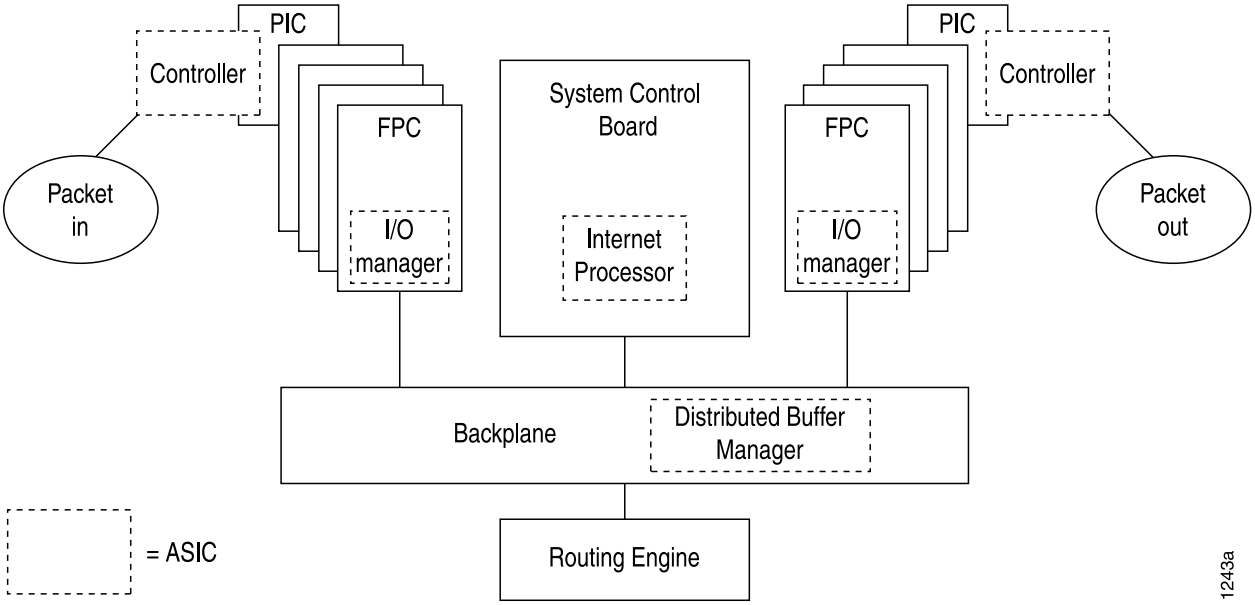
Packet Flow on Juniper Networks M Series Multiservice Edge Routers

On M Series Multiservice Edge Routers, CoS actions are performed in several locations in a Juniper Networks router: the incoming I/O Manager ASIC, the Internet Processor II

Copyright © 2019, Juniper Networks, Inc.

ASIC, and the outgoing I/O Manager ASIC. These locations are shown in Figure 48 on page 562.

Figure 48: M Series Router Packet Forwarding Engine Components and Data Flow



This topic describes the packet flow through the following components in more detail:

- Incoming I/O Manager ASIC on page 562
- Internet Processor ASIC on page 562
- Outgoing I/O Manager ASIC on page 563
- Enhanced CFEB and CoS on M7i and M10i Routers on page 563

Incoming I/O Manager ASIC

When a data packet is passed from the receiving interface to its connected FPC, it is received by the I/O Manager ASIC on that specific FPC. During the processing of the packet by this ASIC, the information in the packet's header is examined by a behavior aggregate (BA) classifier. This classification action associates the packet with a particular forwarding class. In addition, the value of the packet's loss priority bit is set by this classifier. Both the forwarding class and loss priority information are placed into the notification cell, which is then transmitted to the Internet Processor II ASIC.

Internet Processor ASIC

The Internet Processor II ASIC receives notification cells representing inbound data packets and performs route lookups in the forwarding table. This lookup determines the outgoing interface on the router and the next-hop IP address for the data packet. While the packet is being processed by the Internet Processor II ASIC, it might also be evaluated by a firewall filter, which is configured on either the incoming or outgoing interface. This filter can perform the functions of a multifield classifier by matching on multiple elements within the packet and overwriting the forwarding class, loss priority settings, or both within the notification cell. Once the route lookup and filter evaluations are complete,

Copyright © 2019, Juniper Networks, Inc.

the notification cell, now called the result cell, is passed to the I/O Manager ASIC on the FPC associated with the outgoing interface.

Outgoing I/O Manager ASIC

When the result cell is received by the I/O Manager ASIC, it is placed into a queue to await transmission on the physical media. The specific queue used by the ASIC is determined by the forwarding class associated with the data packet. The configuration of the queue itself helps determine the service the packet receives while in this queued state. This functionality guarantees that certain packets are serviced and transmitted before other packets. In addition, the queue settings and the packet's loss priority setting determine which packets might be dropped from the network during periods of congestion.

In addition to queuing the packet, the outgoing I/O Manager ASIC is responsible for ensuring that CoS bits in the packet's header are correctly set before it is transmitted. This rewrite function helps the next downstream router perform its CoS function in the network.

Enhanced CFEB and CoS on M7i and M10i Routers

The Enhanced Compact Forwarding Engine Board (CFEB-E) for the M7i and M10i Multiservice Edge Routers provides additional hardware performance, scaling, and functions, as well as enhanced CoS software capabilities.

The enhanced CoS functions available with the CFEB-E on M7i and M10i routers include:

- Support for 16 forwarding classes and 8 queues
- Support for four loss priorities (medium-high and medium-low in addition to high and low)
- Support for hierarchical policing with tricolor marking, both single-rate tricolor marking (TCM) and two-rate TCM (trTCM)

Related Documentation

- Packet Flow Through the Junos OS CoS Process Overview on page 17
- Packet Flow on MX Series 5G Universal Routing Platforms on page 570
- Packet Flow on Juniper Networks T Series Core Routers on page 656

Copyright © 2019, Juniper Networks, Inc.

Working Around Multifield Classifier Limitations on M Series Routers

On M Series routers (except M120 routers), multifield classifiers are limited such that they cannot classify packets with an output filter match based on the ingress classification that is set with an input filter.

For example, in the following procedure, the filter called **ingress** assigns all incoming IPv4 packets to the **expedited-forwarding** class. The filter called **egress** counts all packets that were assigned to the **expedited-forwarding** class in the **ingress** filter. This configuration does not work on most M Series routers. It works on all other routing platforms, including M120 routers, T Series routers, and MX Series routers.

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

1. Define the ingress filter.

```
[edit]
user@host# edit firewall family inet filter ingress
user@host# set term 1 then forwarding-class expedited-forwarding accept
user@host# set term 2 then accept
```

2. Define the egress filter:

   a. Specify the first term of the egress filter.

```
[edit]
user@host# edit firewall filter egress term 1
user@host# set from forwarding-class expedited-forwarding
user@host# set then count ef
```

   b. Specify the second term of the egress filter.

```
[edit firewall filter egress]
user@host# set term 2 then accept
```

As a workaround, you can configure all of the actions in the ingress filter. For example:

1. Define the ingress filter.

```
[edit]
user@host# edit firewall family inet filter ingress
user@host# set term 1 then forwarding-class expedited-forwarding accept
user@host# set term 1 then forwarding-class expedited-forwarding count ef
user@host# set term 2 then accept
```

Related Documentation

- Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 97

Copyright © 2019, Juniper Networks, Inc.

- Configuring Multifield Classifiers on page 98

Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface

For interfaces with the Frame Relay encapsulation on M120 routers, M320 routers with Enhanced III FPC, M7i and M10i routers with Enhanced Compact Forwarding Engine Board, and MX Series routers, you can set the loss priority of Frame Relay traffic based on the discard eligibility (DE) bit. For each incoming frame with the DE bit containing the class-of-service (CoS) value **0** or **1**, you can configure a Frame Relay loss priority value of low, medium-low, medium-high, or high.

The default Frame Relay loss priority map contains the following settings:

```
loss-priority low code-point 0;
loss-priority high code-point 1;
```

The default map sets the loss priority to low for each incoming frame with the DE bit containing the CoS value **0**. The map sets the loss priority to high for each incoming frame with the DE bit containing the CoS value **1**.

To assign the default Frame Relay DE loss priority map to an interface:

1. Include the **frame-relay-de default** statement at the **[edit class-of-service interfaces** *interface-name* **unit** *logical-unit-number* **loss-priority-maps]** hierarchy level.

   For example:

   ```
   [edit class-of-service interfaces so-1/0/0 unit 0 loss-priority-maps]
   user@host# set frame-relay-de default;
   ```

2. Verify the configuration in operational mode.

   ```
   user@host> show class-of-service loss-priority-map
   Loss-priority-map: frame-relay-de-default, Code point type: frame-relay-de,
   Index: 38
     Code point       Loss Priority
     0                Low
     1                High
   ```

Related Documentation

- Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 361

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 14

Configuring Class of Service on MX Series 3D Universal Edge Routers

- Junos CoS on MX Series 5G Universal Routing Platforms Overview on page 567
- CoS Features and Limitations on MX Series Routers on page 568
- Packet Flow on MX Series 5G Universal Routing Platforms on page 570
- Example of Packet Flow on MX Series 5G Universal Routing Platforms on page 573
- Configuring and Applying IEEE 802.1ad Classifiers on page 575
- Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels on page 576
- Example: Performing Output Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels on page 578
- CoS-Based Interface Counters for IPv4 or IPv6 Aggregate on Layer 2 on page 593
- Enabling a Timestamp for Ingress and Egress Queue Packets on page 594
- Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface on page 595
- Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic on page 596
- Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic on page 597
- Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 597

Junos CoS on MX Series 5G Universal Routing Platforms Overview

The increased demand for sophisticated, media-rich services, the exponential growth of mobile sessions, and the emerging trend of cloud computing require a networking infrastructure that supports massive numbers of subscribers, service types and instances, and bandwidth. A number of features and methods have been developed to address these advanced network requirements, including Junos class of service (CoS). Junos CoS is a set of mechanisms that helps maintain specified service levels for your network by optimizing and prioritizing network traffic so that demand for resources can meet requirements. Use CoS mechanisms to control the allocation of network attributes such as available bandwidth, latency, jitter, packet drop, and bit rate errors so that resources are managed to levels acceptable to your network customers and applications.

Copyright © 2019, Juniper Networks, Inc.

CoS on Juniper Networks MX Series 5G Universal Routing Platforms

MX Series routers are available in a variety of configurations with robust features, including options that provide the level and granularity of the CoS support needed in your network. The MX Series hardware options include several models of Modular Port Concentrators (MPCs), using several different Modular Interface Cards (MICs), and several models of Dense Port Concentrators (DPCs). The MPCs and DPCs provide varying degrees of CoS support.

The MPCs are next-generation line modules for advanced Ethernet services edge and broadband edge networks using high capacity, modular Gigabit Ethernet, 10-Gigabit Ethernet, and 100-Gigabit Ethernet hardware. The MPCs house Packet Forwarding Engines that deliver comprehensive Layer 3 routing (IPv4 and IPv6), Layer 2 switching, inline services, and advanced hierarchical class of service (H-CoS) per MX Series slot. The MPCs can also take advantage of the high performance Junos Trio chipset.

Key CoS features provided by the MPCs include extensive queue management, scheduler hierarchy, shaping, intelligent oversubscription, weighted round robin (WRR), random early detection (RED), and weighted random early detection (WRED).

The DPCs (DPCE-X, DPCE-R, and DPCE-Q) each provide multiple physical interfaces and Packet Forwarding Engines on a single board that performs packet processing and forwarding. Each Packet Forwarding Engine consists of one I-chip for Layer 3 processing and one network processor for Layer 2. DPCE-Qs offer enhanced queuing capabilities and the CoS features of WRR, RED, and WRED.

Related Documentation

- CoS Features and Limitations on MX Series Routers on page 568
- Packet Flow on MX Series 5G Universal Routing Platforms on page 570

CoS Features and Limitations on MX Series Routers

Generally, the Layer 3 CoS hardware capabilities and limitations for Juniper Networks MX Series 5G Universal Routing Platforms are the same as for M Series Multiservice Edge Routers (M120 routers in particular).

In particular, the following scaling and performance parameters apply to MX Series routers:

- 48* classifiers of each type, when subscriber management is enabled
- 32 rewrite tables of each type, when subscriber management is enabled
- Eight queues per port
- 64 WRED profiles
- 100-ms queue buffering for interfaces 1 Gbps and above; 500 ms for all others
- Line-rate CoS features

Copyright © 2019, Juniper Networks, Inc.

NOTE: *Starting with Junos OS Release 16.1R5, Junos OS Release 17.1R3, Junos OS Release 17.2R2, and Junos OS Release 17.3R2, you can configure up to 48 classifiers per family at the **[edit class-of-service classifiers]** hierarchy level when subscriber management is enabled. In earlier releases, you could only configure up to 32 classifiers per family.

For more information about MX Series router CoS capabilities, including software configuration, see "Configuring Hierarchical Schedulers for CoS" on page 342 and "Enhanced Queuing DPC CoS Properties" on page 908.

For Juniper Networks MX Series 5G Universal Routing Platforms, the following restrictions apply:

- You can only use multifield classifiers (but *not* BA classifiers) for IPv4 DSCP bits for virtual private LAN service (VPLS).
- You cannot use BA classifiers for IPv4 DSCP bits for Layer 2 VPNs.
- You cannot use BA classifiers for IPv6 DSCP bits for VPLS.
- You cannot use BA classifiers for IPv6 DSCP bits for Layer 2 VPNs.

On MX Series routers, you can apply classifiers or rewrite rules to an integrated bridging and routing (IRB) interface at the **[edit class-of-service interfaces irb unit *logical-unit-number*]** level of the hierarchy. All types of classifiers and rewrite rules are allowed. These classifiers and rewrite rules are independent of others configured on an MX Series router.

```
[edit class-of-service interfaces]
irb {
  unit logical-unit-number {
    classifiers {
      type (classifier-name | default) family (mpls | all);
    }
    rewrite-rules {
      dscp (rewrite-name | default);
      dscp-ipv6 (rewrite-name | default);
      exp (rewrite-name | default)protocol protocol-types;
      ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
      inet-precedence (rewrite-name | default);
    }
  }
}
```

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

The IRB classifiers and rewrite rules are applied only to the "routed" packets. For logical interfaces that are part of a bridge domain, only IEEE classifiers and IEEE rewrite rules are allowed. Only the listed options are available for rewrite rules on an IRB.

Copyright © 2019, Juniper Networks, Inc.

For dual-tagged bridge domain logical interfaces, you can configure classification based on the inner or outer VLAN tag's IEEE 802.1p bits using the **vlan-tag** statement with the **inner** or **outer** option:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
classifiers {
   ieee-802.1 (classifier-name | default) vlan-tag (inner | outer);
}
```

Also, for dual-tagged bridge domain logical interfaces, you can configure rewrite rules to rewrite the outer or both outer and inner VLAN tag's IEEE 802.1p bits using the **vlan-tag** statement with the **outer** or **outer-and-inner** option:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
rewrite-rules {
   ieee-802.1 (rewrite-rule-name | default) vlan-tag (outer | outer-and-inner);
}
```

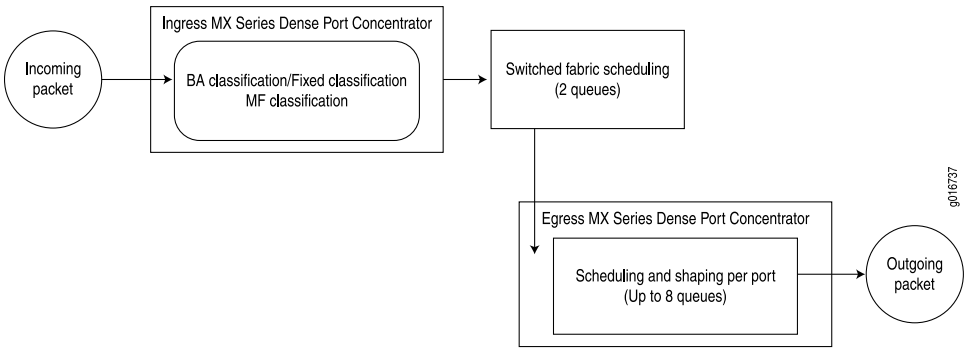
Release History Table

Release	Description
16.1R5	Starting with Junos OS Release 16.1R5, Junos OS Release 17.1R3, Junos OS Release 17.2R2, and Junos OS Release 17.3R2, you can configure up to 48 classifiers per family at the **[edit class-of-service classifiers]** hierarchy level when subscriber management is enabled.

Packet Flow on MX Series 5G Universal Routing Platforms

The CoS architecture for MX Series 5G Universal Routing Platforms, such as the MX960 router, is in concept similar to, but in particulars different from, other routers. The general architecture for MX Series routers is shown in Figure 49 on page 570.Figure 49 on page 570 illustrates packet flow through a Dense Port Concentrator (DPC).

Figure 49: MX Series Router Packet Forwarding and Data Flow



NOTE: All Layer 3 Junos OS CoS functions are supported on the MX Series routers. In addition, Layer 3 CoS capabilities, with the exception of traffic shaping, are supported on virtual LANs (VLANs) that span multiple ports.

Copyright © 2019, Juniper Networks, Inc.

MX Series routers can be equipped with Flexible PIC Concentrators (FPCs) and associated Physical Interface Cards (PICs), Dense Port Concentrators (DPCs), Modular Interface Cards (MICs), Modular Port Concentrators (MPCs), or MPCs with associated MICs. In all cases, the command-line interface (CLI) configuration syntax refers to FPCs, PICs, and ports (***type-fpc/pic/port***).

NOTE: The MX80 router is a single-board router with a built-in Routing Engine and one Packet Forwarding Engine, which can have up to four MICs attached to it. The Packet Forwarding Engine has two "pseudo" Flexible PIC Concentrators (FPC 0 and FPC1). Because there is no switching fabric, the single Packet Forwarding Engine takes care of both ingress and egress packet forwarding.

Fixed classification places all packets in the same forwarding class, or the usual multifield or behavior aggregate (BA) classifications can be used to treat packets differently. BA classification with firewall filters can be used for classification based on IP precedence, DSCP, IEEE, or other bits in the frame or packet header.

However, the MX Series routers can also employ multiple BA classifiers on the same logical interface. The logical interfaces do not have to employ the same type of BA classifier. For example, a single logical interface can use classifiers based on IP precedence as well as IEEE 802.1p. If the CoS bits of interest are on the inner VLAN tag of a dual-tagged VLAN interface, the classifier can examine either the inner or outer bits. (By default, the classification is done based on the outer VLAN tag.)

Internal fabric scheduling is based on only two queues: high and low priority. Strict-high priority queuing is also supported in the high-priority category.

Egress port scheduling supports up to eight queues per port using a form of round-robin queue servicing. The supported priority levels are strict-high, high, medium-high, medium-low, and low. The MX Series router architecture supports both early discard and tail drop on the queues.

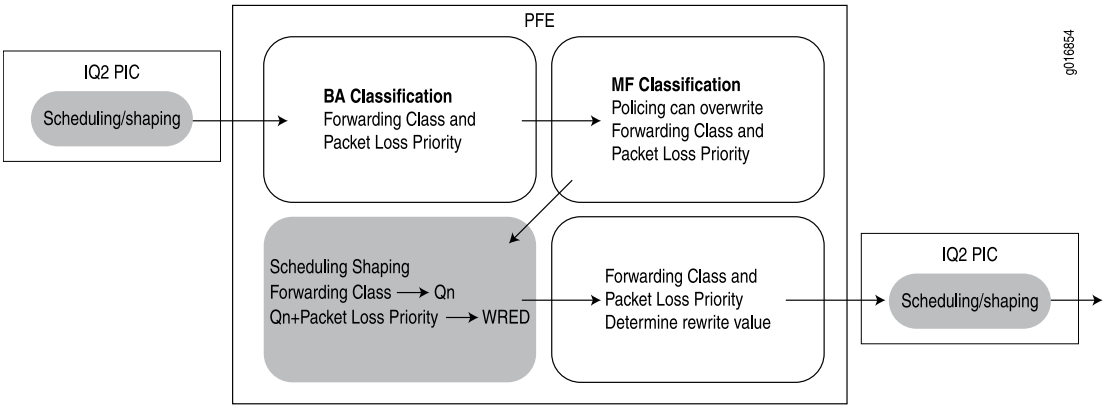
All CoS features are supported at line rate.

The fundamental flow of a packet subjected to CoS is different in the MX Series router with integrated chips than it is in the M Series Multiservice Edge Router and T Series Core Router, which have a different packet-handling architecture.

The way that a packet makes its way through an M Series or T Series router with Intelligent Queuing 2 (IQ2) PICs is shown in Figure 50 on page 572. Note that the per-VLAN scheduling and shaping are done on the PIC whereas all other CoS functions at the port level are performed on the Packet Forwarding Engine.

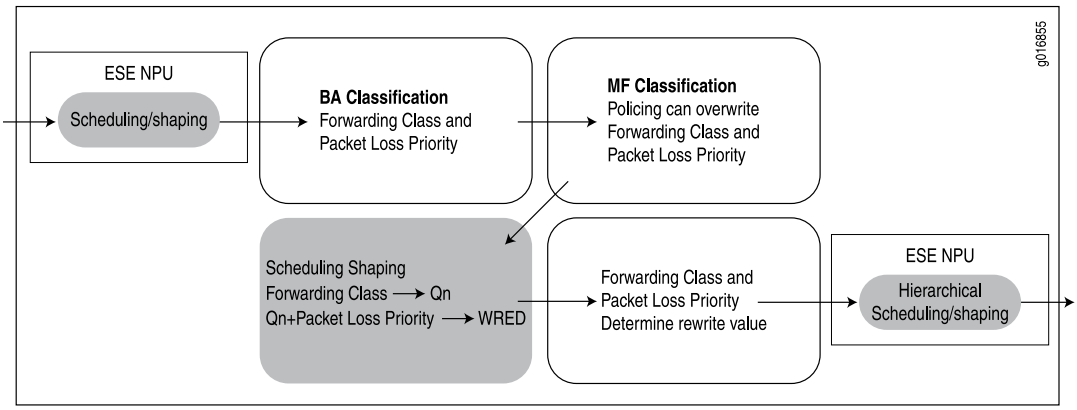
Copyright © 2019, Juniper Networks, Inc.

Figure 50: Packet Handling on the M Series and T Series Routers



The way that a packet makes its way through an MX Series router is shown in Figure 51 on page 572. Note that the scheduling and shaping are done with an integrated architecture along with all other CoS functions. In particular, scheduling and shaping are done on the Ethernet services engine network processing unit (ESE NPU). Hierarchical scheduling is supported on the output side as well as the input side.

Figure 51: Packet Handling on the MX Series Routers



Related Documentation

- Packet Flow Through the Junos OS CoS Process Overview on page 17
- Packet Flow on Juniper Networks M Series Multiservice Edge Routers on page 561
- Example of Packet Flow on MX Series 5G Universal Routing Platforms on page 573
- Packet Flow on Juniper Networks T Series Core Routers on page 656

Copyright © 2019, Juniper Networks, Inc.

Example of Packet Flow on MX Series 5G Universal Routing Platforms

Copyright © 2019, Juniper Networks, Inc.

MX Series routers, especially the MX960 Universal Routing Platform, have several features that differ from the usual CoS features in the Junos OS.

The MX960 router allows fixed classification of traffic. All packets on a logical interface can be put into the same forwarding class. For example:

```
[edit class-of-service interfaces ge-1/0/0 unit 0]
user@host#set forwarding-class af
```

As on other routers, the MX Series routers allow BA classification, the classifying of packets into different forwarding classes (up to eight) based on a value in the packet header. However, MX Series routers allow a mixture of BA classifiers (IEEE 802.1p and others) for logical interfaces on the same port. In the following example, the IEEE classifier is applied to Layer 2 traffic and the Internet precedence classifier is applied to Layer 3 (IP) traffic.

```
[edit class-of-service interfaces ge-0/0/0 unit 0]
user@host#set classifiers ieee-802.1 DOT1P-BA-1
user@host#set classifiers inet-precedence IPPRCE-BA-1
```

The IEEE classifier can also perform BA classification based on the bits of either the inner or outer VLAN tag on a dual-tagged logical interface, as shown in the following example:

```
[edit class-of-service interfaces ge-0/0/0]
user@host#set unit 0 classifiers ieee-802.1 DOT1-BA-1 vlan-tag inner
user@host#set unit 1 classifiers ieee-802.1 DOT1-BA-1 vlan-tag outer
```

NOTE: The example above does not apply to single-tagged packets. The following example shows how to configure the classifier on single-tagged interfaces:

```
[edit class-of-service interfaces ge-0/0/0]
user@host#set unit 0 classifiers ieee-802.1 DOT1-BA-1
```

The default action is based on the outer VLAN tag's IEEE precedence bits.

As on other routers, the BA classification can be overridden with a multifield classifier in the action part of a firewall filter.

Rewrites are handled as on other routers, but MX Series routers support classifications and rewrites for aggregated Ethernet (**ae-**) logical interfaces. MX Series routers also support the use of egress firewall filters for DSCP rewrites for IPv4 and IPv6 packets. For example:

```
[edit firewall family inet]
user@host# set term 1 from destination-address 198.51.100.100/32
user@host# set term 1 then dscp af21
user@host#  set term 2 then accept
```


Copyright © 2019, Juniper Networks, Inc.

On MX Series routers, the 64 classifier limit is a theoretical upper limit. In practice, you can configure 63 classifiers. Three values are used internally by the default IP precedence, IPv6, and EXP classifiers. Two other classifiers are used for forwarding class and queue operations. This leaves 58 classifiers for configuration purposes. If you configure Differentiated Services code point (DSCP) rewrites for MPLS, the maximum number of classifiers you can configure is less than 58.

On MX Series routers, IEEE 802.1 classifier bit rewrites are determined by forwarding class and packet priority, not by queue number and packet priority as on other routers.

Related Documentation

- Packet Flow on Juniper Networks M Series Multiservice Edge Routers on page 561
- Packet Flow on MX Series 5G Universal Routing Platforms on page 570

Configuring and Applying IEEE 802.1ad Classifiers

If you apply an IEEE 802.1 classifier to a logical interface, this classifier takes precedence and is not compatible with any other classifier type. For Juniper Networks MX Series 5G Universal Routing Platform interfaces or IQ2 PICs with IEEE 802.1ad frame formats or EX Series switches, you can set the forwarding class and loss priority for traffic on the basis of the three IEEE 802.1p bits (three bits in either the inner virtual LAN (VLAN) tag or the outer VLAN tag) and the drop eligible indicator (DEI) bit. You can apply the default map or customize one or more of the default values.

You then apply the classifier to the interface on which you configure IEEE 802.1ad frame formats.

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

1. Define the custom IEEE 802.1ad map:

   a. Create the classifier by specifying a name for it and defining it as an IEEE-802.1ad (DEI) classifier.

   ```
   [edit]
   user@host# edit class-of-service classifiers ieee-802.1ad dot1p_dei_class
   ```

   b. Assign the forwarding class and loss priority to the code-point alias.

   ```
   [edit class-of-service classifiers ieee-802.1ad dot1p_dei_class]
   user@host# set forwarding-class best-effort loss-priority low code-points [0000
     1101]
   ```

2. Apply the classifier to the logical interface:

   a. Specify the interface to which you want to apply the classifier.

   ```
   [edit]
   user@host# edit class-of-service interfaces ge-2/0/0 unit 0
   ```

Copyright © 2019, Juniper Networks, Inc.

b. Specify the name of the classifier you want to apply to the interface.

```
[edit class-of-service interfaces ge-2/0/0 unit 0]
user@host# set classifiers ieee-802.1ad dot1p_dei_class
```

3. Verify the custom IEEE 802.1ad map configuration:

```
[edit]
user@host# show
```

```
class-of-service {
  classifiers {
    ieee-802.1ad dot1p_dei_class {
      forwarding-class best-effort {
        loss-priority low code-points [ 0000 1101 ];
      }
    }
  }
}
```

```
class-of-service {
  interfaces {
  ge-2/0/0 {
    unit 0 {
      classifiers {
        ieee-802.1ad dot1p_dei_class;
      }
    }
  }
  }
]
```

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53

Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels

This topic covers the following information:

- Understanding Scheduling and Shaping of Traffic Routed to GRE Tunnels on page 576
- Configuration Overview on page 577
- Configuration Caveats on page 577

Understanding Scheduling and Shaping of Traffic Routed to GRE Tunnels

On MX Series routers running Junos OS Release 12.3R4 or later revisions, 13.2R2 or later revision, or 13.3R1 or later, you can manage CoS scheduling and shaping of traffic routed

Copyright © 2019, Juniper Networks, Inc.

to generic route encapsulation (GRE) tunnel interfaces configured on *MPC1 Q*, *MPC2 Q*, or *MPC2 EQ* modules.

A single egress logical interface can be converted to multiple GRE tunnel interfaces. A GRE tunnel physical interface can support many logical interfaces, but one or more of those logical interfaces might not have an output traffic control profiles attached. If a GRE tunnel logical interface is not attached to an output traffic control profile, the router does not assign the interface a dedicated scheduler. Instead, the interface uses a reserved scheduler intended for all *unshaped tunnel traffic* (traffic entering a GRE tunnel logical interface that does not have an explicit traffic control profile configuration).

Configuration Overview

At GRE tunnel interfaces, the output-traffic-control-profile configuration statement can apply an output traffic scheduling and shaping profile at the physical or logical interface level, while the output-traffic-control-profile-remaining configuration statement can apply an output traffic scheduling and shaping profile for remaining traffic at the physical interface level only. Interface sets (sets of interfaces used to configure hierarchical CoS schedulers on supported Ethernet interfaces) are not supported on GRE tunnel interfaces.

By default—if you do not attach an output traffic control profile to the GRE tunnel physical interface—traffic entering the interface is scheduled and shaped using the default 95/5 scheduler with parameters as specified in the tunnel-services configuration.

If you use an output traffic control profile to configure the shaping rate at the GRE tunnel physical interface, the shaping-rate specified by the attached traffic control profile overrides the **bandwidth** specified as the tunnel services default value.

Configuration Caveats

When configuring hierarchical CoS scheduling and shaping of traffic routed to GRE tunnels, keep the following guidelines in mind:

- You must first configure and commit a hierarchical scheduler on the GRE tunnel physical interface, specifying a maximum of two hierarchical scheduling levels for node scaling. After you commit the hierarchical-scheduler configuration, you can configure scheduling and queuing parameters at the GRE tunnel physical or logical interfaces.
- GRE tunnel interfaces support eight egress queues only. For interfaces on MPC1 Q, MPC2 Q, and MPC2 EQ modules, you can include the max-queues-per-interface **4** statement at the **[edit fpc *slot-number* pic *pic-number*]** hierarchy level to configure four-queue mode for the interface. However, any GRE tunnel interfaces configured on those ports have eight queues.
- Queuing and scheduling calculations include Layer 3 fields. For GRE interfaces, Layer 3 fields include the delivery header (the outer IP header), the 4-byte GRE header, and the payload protocol header and data.

Related Documentation

- Example: Performing Output Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels on page 578
- Per-Unit Queuing and Hierarchical Queuing for MIC and MPC Interfaces on page 978

Copyright © 2019, Juniper Networks, Inc.

- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972

Example: Performing Output Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels

This example shows how to configure a generic routing encapsulation (GRE) tunnel device to perform CoS output scheduling and shaping of IPv4 traffic routed to GRE tunnels. This feature is supported on MX Series routers running Junos OS Release 12.3R4 or later revisions, 13.2R2 or later revision, or 13.3R1 or later, with GRE tunnel interfaces configured on *MPC1 Q*, *MPC2 Q*, or *MPC2 EQ* modules.

- Requirements on page 578
- Overview on page 578
- Configuration on page 579
- Verification on page 590

Requirements

This example uses the following Juniper Networks hardware and Junos OS software:

- Transport network—An IPv4 network running Junos OS Release 13.3.
- GRE tunnel device—One MX80 router installed as an ingress provider edge (PE) router.
- Input and output logical interfaces configurable on two ports of the built-in 10-Gigabit Ethernet Modular Interface Card (MIC):
  - Input logical interface **ge-1/1/0.0** for receiving traffic that is to be transported across the network.
  - Output logical interfaces **ge-1/1/1.0**, **ge-1/1/1.1**, and **ge-1/1/1.2** to convert to GRE tunnel source interfaces **gr-1/1/10.1**, **gr-1/1/10.2**, and **gr-1/1/10.3**.

    For information about interfaces hosted on modules in MX80 routers, see the following topics:

    - *MX5, MX10, MX40, and MX80 Modular Interface Card Description*
    - *MX5, MX10, MX40, and MX80 Port and Interface Numbering*

Overview

In this example, you configure the router with input and output logical interfaces for IPv4 traffic, and then you convert the output logical interface to four GRE tunnel source interfaces. You also install static routes in the routing table so that input traffic is routed to the four GRE tunnels.

Copyright © 2019, Juniper Networks, Inc.

NOTE: Before you apply a traffic control profile with a scheduler-map and shaping rate to a GRE tunnel interface, you must configure and commit a hierarchical scheduler on the GRE tunnel physical interface, specifying a maximum of two hierarchical scheduling levels for node scaling.

Configuration

To configure scheduling and shaping in hierarchical CoS queues for traffic routed to GRE tunnel interfaces configured on MPC1Q, MPC2Q, or MPC2 EQ modules on an MX Series router, perform these tasks:

- Configuring Interfaces, Hierarchical Scheduling on the GRE Tunnel Physical Interface, and Static Routes on page 581
- Measuring GRE Tunnel Transmission Rates Without Shaping Applied on page 584
- Configuring Output Scheduling and Shaping at GRE Tunnel Physical and Logical Interfaces on page 585

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.

Configuring Interfaces, Hierarchical Scheduling on the GRE Tunnel Physical Interface, and Static Routes

```
set chassis fpc 1 pic 1 tunnel-services bandwidth 1g
set interfaces ge-1/1/0 unit 0 family inet address 10.6.6.1/24
set interfaces ge-1/1/1 unit 0 family inet address 10.70.1.1/24 arp 10.70.1.3 mac
00:00:03:00:04:00
set interfaces ge-1/1/1 unit 0 family inet address 10.80.1.1/24 arp 10.80.1.3 mac
00:00:03:00:04:01
set interfaces ge-1/1/1 unit 0 family inet address 10.90.1.1/24 arp 10.90.1.3 mac
00:00:03:00:04:02
set interfaces ge-1/1/1 unit 0 family inet address 10.100.1.1/24 arp 10.100.1.3
mac 00:00:03:00:04:04
set interfaces gr-1/1/10 unit 1 family inet address 10.100.1.1/24
set interfaces gr-1/1/10 unit 1 tunnel source 10.70.1.1 destination 10.70.1.3
set interfaces gr-1/1/10 unit 2 family inet address 10.200.1.1/24
set interfaces gr-1/1/10 unit 2 tunnel source 10.80.1.1 destination 10.80.1.3
set interfaces gr-1/1/10 unit 3 family inet address 10.201.1.1/24
set interfaces gr-1/1/10 unit 3 tunnel source 10.90.1.1 destination 10.90.1.3
set interfaces gr-1/1/10 unit 4 family inet address 10.202.1.1/24
set interfaces gr-1/1/10 unit 4 tunnel source 10.100.1.1 destination 10.100.1.3
set interfaces gr-1/1/10 hierarchical-scheduler
set routing-options static route 10.2.2.0/24 next-hop gr-1/1/10.1
set routing-options static route 10.3.3.0/24 next-hop gr-1/1/10.2
set routing-options static route 10.4.4.0/24 next-hop gr-1/1/10.3
set routing-options static route 10.5.5.0/24 next-hop gr-1/1/10.4
```

Configuring Output Scheduling and Shaping at GRE Tunnel Physical and Logical Interfaces

```
set class-of-service forwarding-classes queue 0 be
set class-of-service forwarding-classes queue 1 ef
set class-of-service forwarding-classes queue 2 af
```

Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service forwarding-classes queue 3 nc
set class-of-service forwarding-classes queue 4 be1
set class-of-service forwarding-classes queue 5 ef1
set class-of-service forwarding-classes queue 6 af1
set class-of-service forwarding-classes queue 7 nc1
set class-of-service classifiers inet-precedence gr-inet forwarding-class be
loss-priority low code-points 000
set class-of-service classifiers inet-precedence gr-inet forwarding-class ef
loss-priority low code-points 001
set class-of-service classifiers inet-precedence gr-inet forwarding-class af
loss-priority low code-points 010
set class-of-service classifiers inet-precedence gr-inet forwarding-class nc
loss-priority low code-points 011
set class-of-service classifiers inet-precedence gr-inet forwarding-class be1
loss-priority low code-points 100
set class-of-service classifiers inet-precedence gr-inet forwarding-class ef1
loss-priority low code-points 101
set class-of-service classifiers inet-precedence gr-inet forwarding-class af1
loss-priority low code-points 110
set class-of-service classifiers inet-precedence gr-inet forwarding-class nc1
loss-priority low code-points 111
set class-of-service interfaces ge-1/1/0 unit 0 classifiers inet-precedence gr-inet
set class-of-service schedulers be_sch transmit-rate percent 30
set class-of-service schedulers ef_sch transmit-rate percent 40
set class-of-service schedulers af_sch transmit-rate percent 25
set class-of-service schedulers nc_sch transmit-rate percent 5
set class-of-service schedulers be1_sch transmit-rate percent 60
set class-of-service schedulers be1_sch priority low
set class-of-service schedulers ef1_sch transmit-rate percent 40
set class-of-service schedulers ef1_sch priority medium-low
set class-of-service schedulers af1_sch transmit-rate percent 10
set class-of-service schedulers af1_sch priority strict-high
set class-of-service schedulers nc1_sch shaping-rate percent 10
set class-of-service schedulers nc1_sch priority high
set class-of-service scheduler-maps sch_map_1 forwarding-class be scheduler be_sch
set class-of-service scheduler-maps sch_map_1 forwarding-class ef scheduler ef_sch
set class-of-service scheduler-maps sch_map_1 forwarding-class af scheduler af_sch
set class-of-service scheduler-maps sch_map_1 forwarding-class nc scheduler nc_sch
set class-of-service scheduler-maps sch_map_2 forwarding-class be scheduler be1_sch
set class-of-service scheduler-maps sch_map_2 forwarding-class ef scheduler ef1_sch
set class-of-service scheduler-maps sch_map_3 forwarding-class af scheduler af_sch
set class-of-service scheduler-maps sch_map_3 forwarding-class nc scheduler nc_sch
set class-of-service traffic-control-profiles gr-ifl-tcp3 guaranteed-rate 5m
set class-of-service traffic-control-profiles gr-ifd-tcp shaping-rate 10m
set class-of-service traffic-control-profiles gr-ifd-tcp-remain shaping-rate 7m
set class-of-service traffic-control-profiles gr-ifd-tcp-remain guaranteed-rate
4m
set class-of-service traffic-control-profiles gr-ifl-tcp1 scheduler-map sch_map_1
set class-of-service traffic-control-profiles gr-ifl-tcp1 shaping-rate 8m
set class-of-service traffic-control-profiles gr-ifl-tcp1 guaranteed-rate 3m
set class-of-service traffic-control-profiles gr-ifl-tcp2 scheduler-map sch_map_2
set class-of-service traffic-control-profiles gr-ifl-tcp2 guaranteed-rate 2m
set class-of-service traffic-control-profiles gr-ifl-tcp3 scheduler-map sch_map_3
set class-of-service interfaces gr-1/1/10 output-traffic-control-profile gr-ifd-tcp
set class-of-service interfaces gr-1/1/10 output-traffic-control-profile-remaining
gr-ifd-remain
set class-of-service interfaces gr-1/1/10 unit 1 output-traffic-control-profile
gr-ifl-tcp1
set class-of-service interfaces gr-1/1/10 unit 2 output-traffic-control-profile
gr-ifl-tcp2
```


Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service interfaces gr-1/1/10 unit 3 output-traffic-control-profile
gr-ifl-tcp3
```

Configuring Interfaces, Hierarchical Scheduling on the GRE Tunnel Physical Interface, and Static Routes

Step-by-Step Procedure

To configure GRE tunnel interfaces (including enabling hierarchical scheduling) and static routes:

1. Configure the amount of bandwidth for tunnel services on the physical interface.

```
[edit]
user@host# set chassis fpc 1 pic 1 tunnel-services bandwidth 1g
```

2. Configure the GRE tunnel device output logical interface.

```
[edit]
user@host# set interfaces ge-1/1/0 unit 0 family inet address 10.6.6.1/24
```

3. Configure the GRE tunnel device output logical interface.

```
[edit]
user@host# set interfaces ge-1/1/1 unit 0 family inet address 10.70.1.1/24 arp 10.70.1.3
  mac 00:00:03:00:04:00
user@host# set interfaces ge-1/1/1 unit 0 family inet address 10.80.1.1/24 arp 10.80.1.3
  mac 00:00:03:00:04:01
user@host# set interfaces ge-1/1/1 unit 0 family inet address 10.90.1.1/24 arp 10.90.1.3
  mac 00:00:03:00:04:02
user@host# set interfaces ge-1/1/1 unit 0 family inet address 10.100.1.1/24 arp
  10.100.1.3 mac 00:00:03:00:04:04
```

4. Convert the output logical interface to four GRE tunnel interfaces.

```
[edit]
user@host# set interfaces gr-1/1/10 unit 1 family inet address 10.100.1.1/24
user@host# set interfaces gr-1/1/10 unit 1 tunnel source 10.70.1.1 destination 10.70.1.3
user@host# set interfaces gr-1/1/10 unit 2 family inet address 10.200.1.1/24
user@host# set interfaces gr-1/1/10 unit 2 tunnel source 10.80.1.1 destination 10.80.1.3
user@host# set interfaces gr-1/1/10 unit 3 family inet address 10.201.1.1/24
user@host# set interfaces gr-1/1/10 unit 3 tunnel source 10.90.1.1 destination 10.90.1.3
user@host# set interfaces gr-1/1/10 unit 4 family inet address 10.202.1.1/24
user@host# set interfaces gr-1/1/10 unit 4 tunnel source 10.100.1.1 destination
  10.100.1.3
```

5. Enable the GRE tunnel interfaces to use hierarchical scheduling.

```
[edit]
user@host# set interfaces gr-1/1/10 hierarchical-scheduler
```

Copyright © 2019, Juniper Networks, Inc.

6. Install static routes in the routing table so that the device routes IPv4 traffic to the GRE tunnel source interfaces.

   Traffic destined to the subnets 10.2.2.0/24, 10.3.3.0/24, 10.4.4.0/24, and 10.5.5.0/24 is routed to the tunnel interfaces at IP addresses 10.70.1.1, 10.80.1.1, 10.90.1.1, and 10.100.1.1, respectively.

```
[edit]
user@host# set routing-options static route 10.2.2.0/24 next-hop gr-1/1/10.1
user@host# set routing-options static route 10.3.3.0/24 next-hop gr-1/1/10.2
user@host# set routing-options static route 10.4.4.0/24 next-hop gr-1/1/10.3
user@host# set routing-options static route 10.5.5.0/24 next-hop gr-1/1/10.4
```

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

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

Results From configuration mode, confirm your configuration by entering the **show chassis fpc 1 pic 1**, **show interfaces ge-1/1/0**, **show interfaces ge-1/1/1**, **show interfaces gr-1/1/10**, and **show routing-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

Confirm the configuration of interfaces, hierarchical scheduling on the GRE tunnel physical interface, and static routes.

```
user@host# show chassis fpc 1 pic 1
tunnel-services {
   bandwidth 1g;
}

user@host# show interfaces ge-1/1/0
unit 0 {
   family inet {
      address 10.6.6.1/24;
   ]
}

user@host# show interfaces ge-1/1/1
unit 0 {
   family inet {
      address 10.70.1.1/24 (
         arp 10.70.1.3 mac 00:00:03:00:04:00;
      }
      address 10.80.1.1/24 {
         arp 10.80.1.3 mac 00:00:03:00:04:01;
      }
      address 10.90.1.1/24 {
         arp 10.90.1.3 mac 00:00:03:00:04:02;
      }
      address 10.100.1.1/24 {
```


Copyright © 2019, Juniper Networks, Inc.

```
        arp 10.100.1.3 mac 00:00:03:00:04:04;
      }
    ]
}

user@host# show interfaces gr-1/1/10
hierarchical-scheduler;
unit 1 {
    tunnel {
        destination 10.70.1.3;
        source 10.70.1.1;
    }
    family inet {
        address 10.100.1.1/24;
    }
}
unit 2 {
    tunnel {
        destination 10.80.1.3;
        source 10.80.1.1;
    }
    family inet {
        address 10.200.1.1/24;
    }
}
unit 3 {
    tunnel {
        destination 10.90.1.3;
        source 10.90.1.1;
    }
    family inet {
        address 10.201.1.1/24;
    }
}
unit 4 {
    tunnel {
        destination 10.100.1.3;
        source 10.100.1.1;
    }
    family inet {
        address 10.202.1.1/24;
    }
}

user@host# show routing-options
static {
    route 10.2.2.0/24 next-hop gr-1/1/10.1;
    route 10.3.3.0/24 next-hop gr-1/1/10.2;
    route 10.4.4.0/24 next-hop gr-1/1/10.3;
    route 10.5.5.0/24 next-hop gr-1/1/10.4;
}
```

Copyright © 2019, Juniper Networks, Inc.

Measuring GRE Tunnel Transmission Rates Without Shaping Applied

Step-by-Step Procedure

To establish a baseline measurement, note the transmission rates at each GRE tunnel source.

1. Pass traffic through the GRE tunnel at logical interfaces **gr-1/1/10.1**, **gr-1/1/10.2**, and **gr-1/1/10.3**.

2. To display the traffic rates at each GRE tunnel source, use the show interfaces queue operational mode command.

   The following example command output shows detailed CoS queue statistics for logical interface gr-1/1/10.1 (the GRE tunnel from source IP address 10.70.1.1 to destination IP address 10.70.1.3).

   ```
   user@host> show interfaces queue gr-1/1/10.1
   Logical interface gr-1/1/10.1 (Index 331) (SNMP ifIndex 4045)
   Forwarding classes: 16 supported, 8 in use
   Egress queues: 8 supported, 8 in use
   Burst size: 0
   Queue: 0, Forwarding classes: be
     Queued:
       Packets              :              31818312                102494 pps
       Bytes                :            6522753960             168091936 bps
     Transmitted:
       Packets              :               1515307                  4879 pps
       Bytes                :             310637935               8001632 bps
       Tail-dropped packets :              21013826                 68228 pps
       RED-dropped packets  :               9289179                 29387 pps
        Low                 :               9289179                 29387 pps
        Medium-low          :                     0                     0 pps
        Medium-high         :                     0                     0 pps
        High                :                     0                     0 pps
       RED-dropped bytes    :            1904281695              48194816 bps
        Low                 :            1904281695              48194816 bps
        Medium-low          :                     0                     0 bps
        Medium-high         :                     0                     0 bps
        High                :                     0                     0 bps
   ...
   ```

   NOTE: This step shows command output for queue 0 (forwarding class be) only.

   The command output shows that the GRE tunnel device transmits traffic from queue **0** at a rate of 4879 pps. Allowing for 182 bytes per Layer 3 packet, preceded by 24 bytes of GRE overhead (a 20-byte delivery header consisting of the IPv4 packet header followed by 4 bytes for GRE flags plus encapsulated protocol type), the traffic rate received at the tunnel destination device is 8,040,592 bps:

   ```
   4879 packets/second X 206 bytes/packet X 8 bits/byte = 8,040,592 bits/second
   ```


Copyright © 2019, Juniper Networks, Inc.

Configuring Output Scheduling and Shaping at GRE Tunnel Physical and Logical Interfaces

Step-by-Step Procedure

To configure the GRE tunnel device with scheduling and shaping at GRE tunnel physical and logical interfaces:

1. Define eight transmission queues.

```
[edit]
user@host# set class-of-service forwarding-classes queue 0 be
user@host# set class-of-service forwarding-classes queue 1 ef
user@host# set class-of-service forwarding-classes queue 2 af
user@host# set class-of-service forwarding-classes queue 3 nc
user@host# set class-of-service forwarding-classes queue 4 be1
user@host# set class-of-service forwarding-classes queue 5 ef1
user@host# set class-of-service forwarding-classes queue 6 af1
user@host# set class-of-service forwarding-classes queue 7 nc1
```

NOTE: To configure up to eight forwarding classes with one-to-one mapping to output queues for interfaces on M120 , M320, MX Series, and T Series routers and EX Series switches, use the queue statement at the [edit class-of-service forwarding-classes] hierarchy level.

If you need to configure up to 16 forwarding classes with multiple forwarding classes mapped to single queues for those interface types, use the class statement instead.

2. Configure BA classifier **gr-inet** that, based on IPv4 precedence bits set in an incoming packet, sets the forwarding class, loss-priority value, and DSCP bits of the packet.

```
[edit]
user@host# set class-of-service classifiers inet-precedence gr-inet forwarding-class
  be loss-priority low code-points 000
user@host# set class-of-service classifiers inet-precedence gr-inet forwarding-class
  ef loss-priority low code-points 001
user@host# set class-of-service classifiers inet-precedence gr-inet forwarding-class
  af loss-priority low code-points 010
user@host# set class-of-service classifiers inet-precedence gr-inet forwarding-class
  nc loss-priority low code-points 011
user@host# set class-of-service classifiers inet-precedence gr-inet forwarding-class
  be1 loss-priority low code-points 100
user@host# set class-of-service classifiers inet-precedence gr-inet forwarding-class
  ef1 loss-priority low code-points 101
user@host# set class-of-service classifiers inet-precedence gr-inet forwarding-class
  af1 loss-priority low code-points 110
user@host# set class-of-service classifiers inet-precedence gr-inet forwarding-class
  nc1 loss-priority low code-points 111
```

Copyright © 2019, Juniper Networks, Inc.

3. Apply BA classifier **gr-inet** to the GRE tunnel device input at logical interface ge-1/1/0.0.

```
[edit]
user@host# set class-of-service interfaces ge-1/1/0 unit 0 classifiers inet-precedence
  gr-inet
```

4. Define a scheduler for each forwarding class.

```
[edit]
user@host# set class-of-service schedulers be_sch transmit-rate percent 30
user@host# set class-of-service schedulers ef_sch transmit-rate percent 40
user@host# set class-of-service schedulers af_sch transmit-rate percent 25
user@host# set class-of-service schedulers nc_sch transmit-rate percent 5
user@host# set class-of-service schedulers be1_sch transmit-rate percent 60
user@host# set class-of-service schedulers be1_sch priority low
user@host# set class-of-service schedulers ef1_sch transmit-rate percent 40
user@host# set class-of-service schedulers ef1_sch priority medium-low
user@host# set class-of-service schedulers af1_sch transmit-rate percent 10
user@host# set class-of-service schedulers af1_sch priority strict-high
user@host# set class-of-service schedulers nc1_sch shaping-rate percent 10
user@host# set class-of-service schedulers nc1_sch priority high
```

5. Define a scheduler map for each of three GRE tunnels.

```
[edit]
user@host# set class-of-service scheduler-maps sch_map_1 forwarding-class be
  scheduler be_sch
user@host# set class-of-service scheduler-maps sch_map_1 forwarding-class ef
  scheduler ef_sch
user@host# set class-of-service scheduler-maps sch_map_1 forwarding-class af
  scheduler af_sch
user@host# set class-of-service scheduler-maps sch_map_1 forwarding-class nc
  scheduler nc_sch
user@host# set class-of-service scheduler-maps sch_map_2 forwarding-class be
  scheduler be1_sch
user@host# set class-of-service scheduler-maps sch_map_2 forwarding-class ef
  scheduler ef1_sch
user@host# set class-of-service scheduler-maps sch_map_3 forwarding-class af
  scheduler af_sch
user@host# set class-of-service scheduler-maps sch_map_3 forwarding-class nc
  scheduler nc_sch
```

6. Define traffic control profiles for three GRE tunnel interfaces.

```
[edit]
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp1 scheduler-map
  sch_map_1
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp1 shaping-rate 8m
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp1 guaranteed-rate
  3m
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp2 scheduler-map
  sch_map_2
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp2 guaranteed-rate
  2m
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp3 scheduler-map
  sch_map_3
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp3 guaranteed-rate
  5m
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp shaping-rate 10m
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp-remain
  shaping-rate 7m
user@host# set class-of-service traffic-control-profiles gr-ifl-tcp-remain
  guaranteed-rate 4m
```

7. Apply CoS scheduling and shaping to the output traffic at the physical interface and logical interfaces.

```
[edit]
user@host# set class-of-service interfaces gr-1/1/10 output-traffic-control-profile
  gr-ifd-tcp
user@host# set class-of-service interfaces gr-1/1/10
  output-traffic-control-profile-remaining gr-ifd-remain
user@host# set class-of-service interfaces gr-1/1/10 unit 1
  output-traffic-control-profile gr-ifl-tcp1
user@host# set class-of-service interfaces gr-1/1/10 unit 2
  output-traffic-control-profile gr-ifl-tcp2
user@host# set class-of-service interfaces gr-1/1/10 unit 2
  output-traffic-control-profile gr-ifl-tcp3
```

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

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

Results From configuration mode, confirm your configuration by entering the **show class-of-service forwarding-classes**, **show class-of-service classifiers**, **show class-of-service interfaces ge-1/1/0**, **show class-of-service schedulers**, **show class-of-service scheduler-maps**, **show class-of-service traffic-control-profiles**, and **show class-of-service interfaces gr-1/1/10** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

Confirm the configuration of output scheduling and shaping at the GRE tunnel physical and logical interfaces.

```
user@host# show class-of-service forwarding-classes
queue 0 be;
queue 1 ef;
queue 2 af;
queue 3 nc;
```

Copyright © 2019, Juniper Networks, Inc.

```
queue 4 be1;
queue 5 ef1;
queue 6 af1;
queue 7 nc1;

user@host# show class-of-service classifiers
inet-precedence gr-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 be1 {
      loss-priority low code-points 100;
   }
   forwarding-class ef1 {
      loss-priority low code-points 101;
   }
   forwarding-class af1 {
      loss-priority low code-points 110;
   }
   forwarding-class nc1 {
      loss-priority low code-points 111;
   }
}

user@host# show class-of-service interfaces ge-1/1/0
unit 0 {
   classifiers {
      inet-precedence gr-inet;
   }
}

user@host# show class-of-service schedulers
be_sch {
   transmit-rate percent 30;
}
ef_sch {
   transmit-rate percent 40;
}
af_sch {
   transmit-rate percent 25;
}
nc_sch {
   transmit-rate percent 5;
}
be1_sch {
   transmit-rate percent 60;
```


Copyright © 2019, Juniper Networks, Inc.

```
    priority low;
}
ef1_sch {
    transmit-rate percent 40;
    priority medium-low;
}
af1_sch {
    transmit-rate percent 10;
    priority strict-high;
}
nc1_sch {
    shaping-rate percent 10;
    priority high;
}

user@host# show class-of-service scheduler-maps
sch_map_1 {
    forwarding-class be scheduler be_sch;
    forwarding-class ef scheduler ef_sch;
    forwarding-class af scheduler af_sch;
    forwarding-class nc scheduler nc_sch;
}
sch_map_2 {
    forwarding-class be scheduler be1_sch;
    forwarding-class ef scheduler ef1_sch;
}
sch_map_3 {
    forwarding-class af scheduler af_sch;
    forwarding-class nc scheduler nc_sch;
}

user@host# show class-of-service traffic-control-profiles
gr-ifl-tcp1 {
    scheduler-map sch_map_1;
    shaping-rate 8m;
    guaranteed-rate 3m;
}
gr-ifl-tcp2 {
    scheduler-map sch_map_2;
    guaranteed-rate 2m;
}
gr-ifl-tcp3 {
    scheduler-map sch_map_3;
    guaranteed-rate 5m;
}
gr-ifd-remain {
    shaping-rate 7m;
    guaranteed-rate 4m;
}
gr-ifd-tcp {
    shaping-rate 10m;
}

user@host# show class-of-service interfaces gr-1/1/10
gr-1/1/10 {
```

Copyright © 2019, Juniper Networks, Inc.

```
    output-traffic-control-profile gr-ifd-tcp;
    output-traffic-control-profile-remaining gr-ifd-remain;
    unit 1 {
      output-traffic-control-profile gr-ifl-tcp1;
    }
    unit 2 {
      output-traffic-control-profile gr-ifl-tcp2;
    }
    unit 3 {
      output-traffic-control-profile gr-ifl-tcp3;
    }
  }
```

Verification

Confirm that the configurations are working properly.

- Verifying That Scheduling and Shaping Are Attached to the GRE Tunnel Interfaces on page 590
- Verifying That Scheduling and Shaping Are Functioning at the GRE Tunnel Interfaces on page 591

Verifying That Scheduling and Shaping Are Attached to the GRE Tunnel Interfaces

Purpose Verify the association of traffic control profiles with GRE tunnel interfaces.

Action Verify the traffic control profile attached to the GRE tunnel physical interface by using the show class-of-service interface **gr-1/1/10 detail** operational mode command.

- `user@host>` **show class-of-service interface gr-1/1/10 detail**

```
Physical interface: gr-1/1/10, Enabled, Physical link is Up
  Type: GRE, Link-level type: GRE, MTU: Unlimited, Speed: 1000mbps
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps

Physical interface: gr-1/1/10, Index: 220
Queues supported: 8, Queues in use: 8
  Output traffic control profile: gr-ifd-tcp, Index: 17721
  Output traffic control profile remaining: gr-ifd-remain, Index: 58414
  Congestion-notification: Disabled

  Logical interface gr-1/1/10.1
    Flags: Point-To-Point SNMP-Traps 0x4000 IP-Header
10.70.1.3:10.70.1.1:47:df:64:0000000000000000 Encapsulation: GRE-NULL
    Gre keepalives configured: Off, Gre keepalives adjacency state: down
    inet  10.100.1.1/24
  Logical interface: gr-1/1/10.1, Index: 331
Object                 Name                  Type                    Index
Traffic-control-profile gr-ifl-tcp1          Output                  17849
Classifier             ipprec-compatibility  ip                         13

  Logical interface gr-1/1/10.2
    Flags: Point-To-Point SNMP-Traps 0x4000 IP-Header
10.80.1.3:10.80.1.1:47:df:64:0000000000000000 Encapsulation: GRE-NULL
```


Copyright © 2019, Juniper Networks, Inc.

```
    Gre keepalives configured: Off, Gre keepalives adjacency state: down
    inet  10.200.1.1/24
  Logical interface: gr-1/1/10.2, Index: 332
Object                 Name                  Type                  Index
Traffic-control-profile gr-ifl-tcp2          Output                17856
Classifier             ipprec-compatibility  ip                       13

  Logical interface gr-1/1/10.3
    Flags: Point-To-Point SNMP-Traps 0x4000 IP-Header
10.90.1.3:10.90.1.1:47:df:64:0000000000000000 Encapsulation: GRE-NULL
    Gre keepalives configured: Off, Gre keepalives adjacency state: down
    inet  10.201.1.1/24
  Logical interface: gr-1/1/10.3, Index: 333
Object                 Name                  Type                  Index
Traffic-control-profile gr-ifl-tcp3          Output                17863
Classifier             ipprec-compatibility  ip                       13
```

Meaning Ingress IPv4 traffic routed to GRE tunnels on the device is subject to CoS output scheduling and shaping.

Verifying That Scheduling and Shaping Are Functioning at the GRE Tunnel Interfaces

Purpose Verify the traffic rate shaping at the GRE tunnel interfaces.

Action

1. Pass traffic through the GRE tunnel at logical interfaces **gr-1/1/10.1**, **gr-1/1/10.2**, and **gr-1/1/10.3**.

2. To verify the rate shaping at each GRE tunnel source, use the show interfaces queue operational mode command.

   The following example command output shows detailed CoS queue statistics for logical interface gr-1/1/10.1 (the GRE tunnel from source IP address 10.70.1.1 to destination IP address 10.70.1.3):

   ```
   user@host> show interfaces queue gr-1/1/10.1
   ```

   ```
   Logical interface gr-1/1/10.1 (Index 331) (SNMP ifIndex 4045)
   Forwarding classes: 16 supported, 8 in use
   Egress queues: 8 supported, 8 in use
   Burst size: 0
   Queue: 0, Forwarding classes: be
     Queued:
       Packets              :              59613061                 51294 pps
       Bytes                :           12220677505              84125792 bps
     Transmitted:
       Packets              :               2230632                  3039 pps
       Bytes                :             457279560                4985440 bps
       Tail-dropped packets :               4471146                  2202 pps
       RED-dropped packets  :              52911283                 46053 pps
        Low                 :              49602496                 46053 pps
        Medium-low          :                     0                     0 pps
        Medium-high         :                     0                     0 pps
   ```

```
    High                 :              3308787                    0 pps
   RED-dropped bytes     :          10846813015             75528000 bps
    Low                  :          10168511680             75528000 bps
    Medium-low           :                    0                    0 bps
    Medium-high          :                    0                    0 bps
    High                 :            678301335                    0 bps
Queue: 1, Forwarding classes: ef
  Queued:
    Packets              :             15344874                51295 pps
    Bytes                :           3145699170             84125760 bps
  Transmitted:
    Packets              :              366115                 1218 pps
    Bytes                :            75053575              1997792 bps
    Tail-dropped packets :              364489                 1132 pps
    RED-dropped packets  :             14614270                48945 pps
    Low                  :             14614270                48945 pps
    Medium-low           :                    0                    0 pps
    Medium-high          :                    0                    0 pps
    High                 :                    0                    0 pps
    RED-dropped bytes    :           2995925350             80270528 bps
    Low                  :           2995925350             80270528 bps
    Medium-low           :                    0                    0 bps
    Medium-high          :                    0                    0 bps
    High                 :                    0                    0 bps
...
```

NOTE: This step shows command output for queue 0 (forwarding class be) and queue 1 (forwarding class ef) only.

Meaning Now that traffic shaping is attached to the GRE tunnel interfaces, the command output shows that traffic shaping specified for the tunnel at logical interface gr-1/1/10.1 (**shaping-rate 8m** and **guaranteed-rate 3m**) is honored.

- For queue **0**, the GRE tunnel device transmits traffic at a rate of 3039 pps. The traffic rate received at the tunnel destination device is 5,008,272 bps:

```
3039 packets/second X 206 bytes/packet X 8 bits/byte = 5,008,272 bits/second
```

- For queue **0**, the GRE tunnel device transmits traffic at a rate of 1218 pps. The traffic rate received at the tunnel destination device is 2,007,264 bps:

```
1218 packets/second X 206 bytes/packet X 8 bits/byte = 2,007,264 bits/second
```

Compare these statistics to the baseline measurements taken without traffic shaping, as described in "Measuring GRE Tunnel Transmission Rates Without Shaping Applied" on page 584.

 Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels on page 576
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

CoS-Based Interface Counters for IPv4 or IPv6 Aggregate on Layer 2

Beginning with Junos OS Release 14.1, Layer 2 CoS-based traffic metering is available for MX series routers with MPCs and MX80 routers. It can be used in greenfield deployments or as a replacement for term-matching counters configured via firewall filters. (Term-matching counters configured via firewall filters can have several drawbacks, including a one-filter-per-family limit, the inclusion of overhead bytes, and less operational efficiency than CoS-based counters). With CoS-based counters, a single aggregate counter per forwarding class can be used for inet and inet6 flows. Both bytes and packet total are counted. Note that flow rates are not measured, and forwarding-class accounting for host-bound traffic is not supported.

You can configure the counters with any or all of the following parameters:

- Logical | physical interfaces
- IPv4 | IPv6 traffic
- Unicast | multicast traffic
- Ingress | egress flows

CoS-based interface counters are highly accurate and can be configured to exclude overhead bytes (such as protocol encapsulations) so end-customer packets can be differentiated from other traffic. At ingress only packets forwarded to the fabric are counted, and at egress only packets forwarded to the WAN are counted. In other words, **forwarding-class accounting** applies to transit traffic only, not host-generated or host-bound traffic. Non-relevant network protocols such as ARP, BFD, and EOAM, as well as dropped packets, are not counted.

To support native interface counters, a new CLI option, **enhanced**, is introduced under **forwarding-class accounting** at both the physical and logical interface levels:

```
interfaces {
  interface-name{
    forwarding-class-accounting {
      enhanced {
        overhead-bytes overhead-value(;
        traffic-type (ucast | mcast);
        family (ipv4 | ipv6 | both );
        direction (ingress | egress | both);
      }
    }
  }
}
```

To view additional counter details, run the following **show** commands:

Copyright © 2019, Juniper Networks, Inc.

- **show interfaces forwarding-class-counters** *interface-name*
- **show class-of-service interface** *interface-name* **comprehensive**
- **show class-of-service interface** *interface-name* **detail**

The **comprehensive** option shows both forwarding-class accounting parameters and the forwarding-class counter, whereas **detail** shows only the forwarding-class accounting parameters. It does not display counters for each forwarding class.

Release History Table

Release	Description
14.1	Beginning with Junos OS Release 14.1, Layer 2 CoS-based traffic metering is available for MX series routers with MPCs and MX80 routers.

Related Documentation

- forwarding-class-accounting on page 1136
- enhanced on page 1100
- show class-of-service interface on page 1364

Enabling a Timestamp for Ingress and Egress Queue Packets

Beginning with Junos OS Release 16.1, you can enable a packet timestamp feature to record the time at which the last packet is enqueued for CoS ingress and egress queues. Timestamps are enabled and reported per FPC. When the feature is enabled, the Packet Forwarding Engine begins collection the timestamp for all ingress and egress queue counters on the FPC. By default, packet timestamp information is not collected.

To activate packet timestamp collection for CoS ingress and egress queues:

- Enable the timestamp on the desired FPC.

```
[edit chassis fpc slot-number traffic-manager]
user@host# set packet-timestamp enable
```

NOTE: When you enable or disable the packet timestamp for an FPC that is already up, the FPC is automatically rebooted when you commit the changes. The action takes effect when the FPC is back up.

NOTE: For aggregated Ethernet interfaces, enable the packet timestamp on all FPCs that have an aggregated Ethernet leg. When you display the queue statistics for the interface, the timestamps for all the legs are shown.

Copyright © 2019, Juniper Networks, Inc.

The following commands display the collected timestamps in the Last-packet enqueued field:

- show interfaces queue **both-ingress-egress** ***interface-name***
- show interfaces queue ***interface-name***
- show interfaces queue ***interface-name*** **aggregate**

Release History Table

Release	Description
16.1	Beginning with Junos OS Release 16.1, you can enable a packet timestamp feature to record the time at which the last packet is enqueued for CoS ingress and egress queues.

Related Documentation

- CoS-Based Interface Counters for IPv4 or IPv6 Aggregate on Layer 2 on page 593
- Packet Flow on MX Series 5G Universal Routing Platforms on page 570

Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface

For interfaces with the Frame Relay encapsulation on M120 routers, M320 routers with Enhanced III FPC, M7i and M10i routers with Enhanced Compact Forwarding Engine Board, and MX Series routers, you can set the loss priority of Frame Relay traffic based on the discard eligibility (DE) bit. For each incoming frame with the DE bit containing the class-of-service (CoS) value **0** or **1**, you can configure a Frame Relay loss priority value of low, medium-low, medium-high, or high.

The default Frame Relay loss priority map contains the following settings:

```
loss-priority low code-point 0;
loss-priority high code-point 1;
```

The default map sets the loss priority to low for each incoming frame with the DE bit containing the CoS value **0**. The map sets the loss priority to high for each incoming frame with the DE bit containing the CoS value **1**.

To assign the default Frame Relay DE loss priority map to an interface:

1. Include the **frame-relay-de default** statement at the **[edit class-of-service interfaces** ***interface-name*** **unit** ***logical-unit-number*** **loss-priority-maps]** hierarchy level.

   For example:

   ```
   [edit class-of-service interfaces so-1/0/0 unit 0 loss-priority-maps]
   user@host# set frame-relay-de default;
   ```

2. Verify the configuration in operational mode.

   ```
   user@host> show class-of-service loss-priority-map
   ```

Copyright © 2019, Juniper Networks, Inc.

```
Loss-priority-map: frame-relay-de-default, Code point type: frame-relay-de,
Index: 38
  Code point       Loss Priority
  0                Low
  1                High
```

Related Documentation

- Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 361

Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic

This topic provides a summary of the configuration for setting the IEEE 802.1p field in the Ethernet frame header for host outbound traffic (control plane traffic). You can set a global value for the priority code point that applies to all host outbound traffic. Additionally, or alternatively, you can specify that rewrite rules are applied to all host outbound traffic on egress logical interfaces. These are rules that have been previously configured to set the IEEE 802.1p field for data traffic on those interfaces.

Configuration of 802.1p bits is supported only on the following hardware and software components:

- EX Series switches
- MX Series 5G Universal Routing Platforms
- Enhanced Queuing DPCs
- MPCs
- Junos OS Release 12.3 or later

To configure the IEEE 802.1p field settings:

1. (Optional) Specify a global default value for the IEEE 802.1p field for all host outbound traffic.

   See "Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic" on page 597.

2. (Optional) Specify that the IEEE 802.1p rewrite rules for the egress logical interfaces are applied to all host outbound traffic on those interfaces.

   See "Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface" on page 597.

Related Documentation

- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388

Copyright © 2019, Juniper Networks, Inc.

Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic

This topic describes how to configure a global default value for the IEEE 802.1p field for all host outbound traffic on MX Series routers and EX Series switches.

To configure a global default value for the IEEE 802.1p field:

- Specify the value.

```
[edit class-of-service host-outbound-traffic ieee-802.1]
user@host# set default value
```

For example, specify that a value of 010 is applied to all host outbound traffic:

```
[edit class-of-service host-outbound-traffic ieee-802.1]
user@host# set default 010
```

Related Documentation

- Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic on page 596
- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388

Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface

This topic describes how to apply rewrite rules for egress logical interfaces to the IEEE 802.1p field for all host outbound traffic on those interfaces on MX Series routers and EX Series switches.

This task requires separately configured rewrite rules that map packet loss priority information to the code point value in the 802.1p field for data traffic on egress logical interfaces. See "Rewriting Packet Headers to Ensure Forwarding Behavior" on page 388.

To configure the rewrite rules:

1. Configure the CoS rewrite rules to map the forwarding class to the desired value for the 802.1p field.

   See "Configuring Rewrite Rules" on page 391.

2. Associate the rewrite rules to the desired egress logical interfaces.

   See "Applying Rewrite Rules to Output Logical Interfaces" on page 401.

3. (Optional) Configure the forwarding class for host outbound traffic. Do not configure this forwarding class if you want to use the default forwarding class assignment (input classification).

   See "Overriding the Input Classification" on page 250.

Copyright © 2019, Juniper Networks, Inc.

To configure the rewrite rules to apply to the host outbound traffic IEEE 802.1p field:

- Configure the rewrite rules.

```
[edit class-of-service host-outbound-traffic ieee-802.1]
user@host# set rewrite-rules
```

NOTE: Enabling IEEE 802.1p rewrite rules for host outbound traffic on a DPC without creating any corresponding IEEE 802.1p rewrite rules on a logical interface on the DPC causes the IEEE 802.1p code point to be automatically set to 000 for all host generated traffic that exits that logical interface.

```
[edit class-of-service]
rewrite-rules {
   ieee-802.1 rewrite_foo {
      forwarding-class network-control {
         loss-priority low code-point 101;
      }
   }
}
interfaces {
   ge-1/0/0 {
      unit 100 {
         rewrite-rules {
            ieee-802.1 rewrite_foo vlan-tag outer-and-inner;
         }
      }
   }
}
host-outbound-traffic {
   forwarding-class network-control;
}
host-outbound-traffic {
   ieee-802.1 {
      rewrite-rules;
   }
}
```

Related Documentation

- Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic on page 596
- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 15

Configuring Class of Service on PTX Series Packet Transport Routers

- CoS Features and Limitations on PTX Series Routers on page 599
- CoS Feature Differences Between PTX Series Packet Transport Routers and T Series Routers on page 601
- Understanding Scheduling on PTX Series Routers on page 604
- Understanding Virtual Output Queues on PTX Series Packet Transport Routers on page 607
- Example: Configuring Excess Rate for PTX Series Packet Transport Routers on page 617
- Identifying the Source of RED Dropped Packets on PTX Series Routers on page 624
- Configuring Virtual LAN Queuing and Shaping on PTX Series Routers on page 631
- Example: Configuring Virtual LAN Queuing and Shaping in PTX Series Packet Transport Routers on page 634
- Example: Configuring Strict-Priority Scheduling on a PTX Series Router on page 637
- Understanding CoS CLI Configuration Statements on PTX Series Routers on page 645

CoS Features and Limitations on PTX Series Routers

Table 63 on page 599 summarizes CoS features and limitations on PTX Series Packet Transport Routers.

The following table lists the CoS features supported on the PTX Series router, as well as the limitations relevant to the PTX Series router. Note that this list is a subset of the overall CoS feature set.

Table 63: CoS Features and Limitations on PTX Series Routers

CoS Feature	Comments
Classifiers	

Copyright © 2019, Juniper Networks, Inc.

Table 63: CoS Features and Limitations on PTX Series Routers (continued)

CoS Feature	Capacity	Comments
Maximum number per PFE	64	L2 classifiers (sum of IEEE-802.1p and IEEE-802.1ad cannot exceed 32) DSCP and IP precedence classifiers (sum of DSCP and IP precedence classifiers cannot exceed 32) DSCP IPv6 classifiers MPLS EXP classifiers
dscp	Yes	DSCP and IP precedence classifiers cannot be configured on the same logical interface.
dscp-ipv6	Yes	Separate classifiers can be applied for IPv4 and IPv6 packets per logical interface.
ieee-802.1p	Yes	You can associate IEEE-802.1p with any other type of classifier on the same logical interface. For L3 packets, an L3 classifier takes precedence over an IEEE classifier. NOTE: IEEE classifiers are not supported on PTX10001-20C routers.
inet-precedence	Yes	
mpls-exp	Yes	NOTE: MPLS EXP classifiers are not supported on PTX10001-20C routers.
Loss priorities based on the Frame Relay discard eligible (DE) bit	No	
Drop Profiles		
Maximum number	32	You can configure up to 32 drop profiles in the PTX chassis.
Per queue	Yes	
Per loss priority	Yes	NOTE: Packet loss priority **medium-low** is not supported on PTX1000 routers.
Per Transmission Control Protocol (TCP) bit	No	
Policing		
Traffic policing	Yes	
Two-rate tricolor marking (TCM)	Yes	NOTE: Tri-color marking is not supported on PTX1000 routers.
Queuing		
Priority	Yes (4)	
Per-queue output statistics	Yes	Red-dropped counters are not maintained per drop precedence. Also tail drop counters always show zero because packets are always dropped by the RED algorithm.

Copyright © 2019, Juniper Networks, Inc.

Table 63: CoS Features and Limitations on PTX Series Routers (continued)

CoS Feature	Capacity	Comments
transmit-rate percent	Yes	Percentage transmit rate for a scheduler has the range 1 through 100 percent. The range is 0 through 100 percent for M, MX and T Series routers and EX Series switches; and 0 through 200 percent for the SONET/SDH OC48/STM16 IQE PIC. On PTX Series routers, unconfigured interfaces are equivalent to **percent 0**.
Rewrite Markers		
Maximum number per PFE	64	The sum of L2 and L3 rewrite rules cannot exceed 64. NOTE: Rewrite rules are not supported on PTX10001-20C routers.
dscp	Yes	
dscp-ipv6	Yes	
ieee-802.1	Yes	L2 and L3 rewrites can be applied to the same packet simultaneously.
inet-precedence	No	
mpls-exp	Yes	

CoS Feature Differences Between PTX Series Packet Transport Routers and T Series Routers

This topic provides a list of class-of-service features available on PTX Series routers and compares them with class-of-service features on T Series routers.

Classifiers

- T Series routers support VRF table labels for Layer 3 VPNs. On PTX Series routers, this feature is not supported.
- On T Series routers, IEEE 802.1 classifiers cannot coexist with Layer 3 classifiers. On PTX Series routers, these classifiers can coexist.
- On T Series routers, IEEE classifiers are supported on Ethernet IQ, IQ2, and IQ2-E interfaces. These interfaces have the flexibility of classifying traffic based on inner or outer VLAN tags. On PTX Series routers, IEEE classification is always based on outer VLAN tags.

Rewrite

- PTX Series routers do not support rewrite of both **exp** and **inet-precedence** fields using:
  - exp protocol mpls-any
  - exp protocol mpls-inet-both
  - exp protocol mpls-inet-both-non-vpn

Copyright © 2019, Juniper Networks, Inc.

- On T Series routers, DSCP rewrite and DSCP IPv6 rewrite are not supported for the MPLS protocol. PTX Series routers support rewrite of both DSCP and DSCP IPv6 for protocol MPLS.
- PTX Series routers support Layer 2 rewrite of 802.1p and 802.1ad, to either the outer VLAN tag, or both outer and inner VLAN tags.

Forwarding Class

- On T Series routers, you can override the default fabric priority queuing of egress traffic by including the **priority** statement at the following hierarchy level:

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

  On PTX Series routers, fabric priority queuing is not supported; therefore, the **priority** statement for **forwarding-classes** is not supported.

Tricolor Marking

- On T Series routers, the **copy-plp-all** statement must be configured to support tricolor marking. On PTX Series routers, tricolor marking is enabled by default.

Schedulers

- T Series routers, which use egress queuing architecture, support chassis and fabric schedulers. Alternatively, PTX Series routers support a virtual output queuing (VOQ) architecture and the fabric schedulers utilize the CoS scheduling parameters to configure the fabric schedulers. There is not a separate configuration for the fabric schedulers on PTX Series routers. With the VOQ architecture, packets are queued and dropped on ingress during congestion.
- On T Series routers, high-priority queues have precedence to acquire excess bandwidth and might consume all excess bandwidth. On PTX Series routers, excess bandwidth is shared based on the ratio of the configured transfer rate. Therefore, all priority queues get a share of excess bandwidth.
- On T Series routers, strict-high-priority queues and high-priority queues are assigned the same hardware priority. On PTX Series routers, strict-high-priority queues and high-priority queues are assigned different hardware priorities. Strict-high-priority can starve other queues if a rate limiter is not applied on PTX Series routers.
- On T Series routers, if a strict-high-priority queue is oversubscribed, it can block all other queues except high-priority queues. On PTX Series routers, if a strict-high-priority queue is oversubscribed, it can block all other queues including high-priority queues.

  To restrict the bandwidth of strict-high priority queues on PTX Series routers, use the **transmit-rate rate-limit** configuration statement.
- On both T Series routers and PTX Series routers, if a strict-high-priority queue is oversubscribed and results in oversubscription of the guaranteed bandwidth, the distribution of bandwidth that is not taken up by strict-high-priority queues is undetermined. T Series routers and PTX Series routers distribute this unused bandwidth differently.

Copyright © 2019, Juniper Networks, Inc.

Buffer Size and Latency

- On T Series routers, memory allocation dynamic (MAD) is enabled by default and can be disabled. PTX Series routers do not support MAD.
- On T Series routers, the maximum delay bandwidth buffering configured per queue is 50 ms. On PTX Series routers, the maximum delay bandwidth buffering configured per queue is 100 ms.
- On T Series routers, the maximum latency associated with a packet is fairly consistent and independent of the number of sources sending the traffic to an interface. On PTX Series routers, over-provisioning is possible. When traffic is sent from multiple Packet Forwarding Engines (PFEs), the latency can be about 10 percent to 15 percent higher than when traffic is sent from one PFE due to the PTX dynamically adjusting buffers from multiple PFEs for a Virtual Output Queue (VOQ). Regardless, the average and minimum latency on a PTX Series router should be much smaller than on a T Series router. The maximum latency on a PTX Series router is controlled by the buffer size that you configure for the VOQ, much like configuring a buffer size on a T Series router.
- On T Series routers, a high-priority queue has lower latency than a low priority queue with the same configured transfer rate and same offered load. On PTX Series routers, there is no latency difference.

Drop Profile

- The Queuing and Memory Interfaces ASIC does not support drop-profile assignments for a queue based on the protocol. As a consequence, the **protocol** option for the **drop-profile-map** configuration statement is treated as **protocol any**.

Interface Queue Statistics

- On T Series routers, transmitted byte counters are computed using Layer 3 packet length. On PTX Series routers, transmitted byte counters are computed using the full Layer 2 overhead (including all L2 encapsulation and CRC) plus 12 for the inter-packet gap plus 8 for the preamble.
- On T Series routers, the tail-dropped counters and the RED-dropped counters are displayed separately in the output of the **show interfaces queue** command. On PTX Series routers, tail-dropped counters are always zero. All the packet drops are shown as RED-dropped in the **show interfaces voq** output.

Related Documentation

- Understanding CoS CLI Configuration Statements on PTX Series Routers on page 645

Copyright © 2019, Juniper Networks, Inc.

Understanding Scheduling on PTX Series Routers

This topic covers class of service packet scheduling for interfaces on PTX Series routers:

- Output Queue Priorities Supported by the Junos OS CLI on PTX Series Routers on page 604
- Scheduling Processes on PTX Series Routers on page 604
- Strict-Priority and Scheduling Processes on PTX Series Routers on page 605

Output Queue Priorities Supported by the Junos OS CLI on PTX Series Routers

Output queues on the PTX Series interface hardware support these values for the queue priority—high, medium, low, and excess. The Junos OS supports five queue priority levels: **strict-high**, **high**, **medium-high**, **medium-low**, and **low**.

NOTE: If a strict-high-priority queue is constantly loaded to 100 percent of traffic capacity, other queues are starved. Queue starvation can cause the interface hardware to generate interrupts.

This starvation can be alleviated by using a rate-limiter on the strict-high queues.

Scheduling Processes on PTX Series Routers

Physical interfaces on PTX Series routers support two mutually exclusive scheduling processes:

- *normal scheduling* (default mode) — A queue's transmit-rate is used to determine whether it is operating within the *guaranteed region* or in the *excess region*.

  Within the *guaranteed region* (transmit-rates credits are positive), the scheduler uses the transmit rates to decide the bandwidth allocation. Queues that are at priority-level **low** or higher and have transmit-rate credits are serviced first by priority order and then within a priority level using packet round robin algorithm.

  Within the *excess region* (for all queues in which **transmit-rate** credits are negative), CoS queues are selected based on the weighted round-robin (WRR) algorithm. If a queue does not have **excess-rate** configured, then its weight is set to 1.

  If a queue is not configured with a **transmit-rate** statement (to specify the transmit rate or percentage of transmission capacity), then it will not be scheduled at **priority** level and only at **excess** level (unless priority is **strict-high**).

  A queue must have a transmit-rate assigned to it in order to be scheduled at priority. excess-rate is used only to determine its weight when scheduled in the excess-region.

  If multiple queues are in the excess region (queue priority **low**) and the **excess-rate** statement is used, then those queues are selected using the WRR algorithm.

Copyright © 2019, Juniper Networks, Inc.

The **remainder** keyword can be applied to transmit-rate as well as buffer size statements. This will cause the remaining portion of the specified resource to be assigned to the queue. The remaining resources for transmit-rate are the sum of all queues that specified a transmit-rate subtracted from the total transmit-rate available. If there are multiple queues assigned with the **remainder** keyword, the remainder of the resource is evenly divided amongst those queues.

There are two keywords that can be applied to the **transmit-rate** statement in order to limit the rate of a queue: **rate-limit** and **exact**, where **rate-limit** can be applied only to the **strict-high** queue and **exact** can be used for all other queues. The two keywords behave identically in that queues can transmit only up to the specified rate. All of their transmission will be scheduled at their configured priority level and they will never be scheduled at **excess-priority** level.

BEST PRACTICE: **The rate-limit option of the transmit-rate configuration statement is allowed only on the strict-high queue. We recommend that you configure rate limit on strict-high queues because the other queues might not meet their guaranteed bandwidths.**

Non strict-high queues can use the option exact to place a maximum transmit-rate limit on them, which is equivalent to rate-limit.

- *strict-priority scheduling* — Queues are processed in strict-priority order. There is no concept of guaranteed region and excess region. The packet scheduler is always operating in the guaranteed region, with the exception of priority **low**, which is always assigned to the excess priority level. The configured **transmit-rate** does not affect how the queue is serviced because packets are processed in order of queue priority. Among queues that are configured **low** priority, if **excess-rate** weights are configured, they are used by the hardware to perform WRR. Queues that are mapped to the same hardware priority or that have the same configured priority other than **low** are serviced in a packet round-robin fashion.

  The queues are serviced in strict-priority order until they have reached their transmit-rate (that is, their guaranteed rate) and then are demoted to the excess level.

  NOTE: **The rate-limit and exact options of the transmit-rate configuration statement have no effect when strict-priority scheduling is configured.**

  To configure strict-priority scheduling for a physical interface on a PTX Series router, include the strict-priority-scheduler and scheduler-map ***map-name*** configuration statements in the traffic control profile you associate with an output interface.

Strict-Priority and Scheduling Processes on PTX Series Routers

Table 64 on page 606 shows the different configurations available for Junos Priority Scheduler Modes, including those for Strict-Priority and Enhanced-Priority Modes. Table 64 on page 606 also shows how the output queue priority values in the Junos OS map to the output queue priorities supported by physical interfaces on PTX Series routers,

Copyright © 2019, Juniper Networks, Inc.

and the scheduling action taken. Starting in Junos OS Release 17.4, the table shows differences for normal scheduling when **strict-high** is not configured, and for strict-priority scheduling.

Table 64: Strict-Priority and Scheduling Processes on PTX Series Routers

Junos Priority	Scheduler Mode	Normal			Strict Priority Scheduler
	Chassis Knob	enhanced-priority-mode		no-enhanced-priority-mode	*
	Strict-High Config	No	Yes	*	*
strict-high		–	High	High	High
high		High	Medium	High	High
medium-high		Medium	Low	Medium	Medium
medium-low		Medium	Low	Medium	Low
low		Low	Low	Low	Excess

NOTE: Packet scheduling is strict priority round-robin while the virtual output queues are in the guaranteed region.

After the virtual output queues consume their guaranteed credits, they are demoted to excess-priority scheduling, which is weighted round-robin.

The only exception is the strict-high priority, which is always scheduled as strict-high-priority.

Release History Table

Release	Description
17.4	Starting in Junos OS Release 17.4, the table shows differences for normal scheduling when **strict-high** is not configured, and for strict-priority scheduling.

Related Documentation

- Understanding CoS CLI Configuration Statements on PTX Series Routers on page 645
- CoS Feature Differences Between PTX Series Packet Transport Routers and T Series Routers on page 601
- How Schedulers Define Output Queue Properties on page 254
- Example: Configuring Strict-Priority Scheduling on a PTX Series Router on page 637
- Example: Configuring Excess Rate for PTX Series Packet Transport Routers on page 617

Copyright © 2019, Juniper Networks, Inc.

- show interfaces voq on page 1468
- strict-priority-scheduler on page 1290
- traffic-control-profiles on page 1305
- transmit-rate on page 1311

Understanding Virtual Output Queues on PTX Series Packet Transport Routers

This section describes the virtual output queue (VOQ) architecture on PTX Series Packet Transport Routers and includes the following topics:

- Introduction to Virtual Output Queues on PTX Series Packet Transport Routers on page 607
- Understanding How VOQ Works on PTX Series Routers on page 611
- Fabric Scheduling and Virtual Output Queues on PTX Series Routers on page 613
- Understanding the Packet Forwarding Engine Fairness and Virtual Output Queue Process on page 615

Introduction to Virtual Output Queues on PTX Series Packet Transport Routers

This topic introduces the virtual output queue (VOQ) architecture on PTX Series Packet Transport routers and how it operates with the configurable class-of-service (CoS) components on PTX Series routers.

Junos OS and PTX Series hardware CoS features use *virtual output queues* on the *ingress* to buffer and queue traffic for each egress output queue. The PTX Series router supports up to eight egress output queues per output port (physical interface).

The traditional method of forwarding traffic through a router is based on buffering ingress traffic in input queues on ingress interfaces, forwarding the traffic across the fabric to output queues on egress interfaces, and then buffering traffic again on the output queues before transmitting the traffic to the next hop. The traditional method of queueing packets on an ingress port is storing traffic destined for different egress ports in the same input queue (buffer).

During periods of congestion, the router might drop packets at the egress port, so the router might spend resources transporting traffic across the switch fabric to an egress port, only to drop that traffic instead of forwarding it. And because input queues store traffic destined for different egress ports, congestion on one egress port could affect traffic on a different egress port, a condition called head-of-line blocking (HOLB).

Virtual output queue (VOQ) architecture takes a different approach:

- Instead of separate physical buffers for input and output queues, the PTX Series router uses the physical buffers on the ingress pipeline of each Packet Forwarding Engine to store traffic for every egress port. Every output queue on an egress port has buffer storage space on every ingress pipeline on all of the Packet Forwarding Engines on the router. The mapping of ingress pipeline storage space to output queues is 1-to-1, so each output queue receives buffer space on each ingress pipeline.

Copyright © 2019, Juniper Networks, Inc.

- Instead of one input queue containing traffic destined for multiple different output queues (a one-to-many mapping), each output queue has a dedicated VOQ comprised of the input buffers on each Packet Forwarding Engine that are dedicated to that output queue (a 1-to-1 mapping). This architecture prevents communication between any two ports from affecting another port.
- Instead of storing traffic on a physical output queue until it can be forwarded, a VOQ does not transmit traffic from the ingress port across the fabric to the egress port until the egress port has the resources to forward the traffic. A VOQ is a collection of input queues (buffers) that receive and store traffic destined for one output queue on one egress port. Each output queue on each egress port has its own dedicated VOQ, which consists of all of the input queues that are sending traffic to that output queue.

A VOQ is a collection of input queues (buffers) that receive and store traffic destined for one output queue on one egress port. Each output queue on each egress port has its own dedicated VOQ, which consists of all of the input queues that are sending traffic to that output queue.

VOQ Architecture

A VOQ represents the ingress buffering for a particular output queue. Each of the Packet Forwarding Engines in the PTX Series router uses a specific output queue. The traffic stored on the Packet Forwarding Engines comprises the traffic destined for one particular output queue on one port, and is the VOQ for that output queue.

A VOQ is distributed across all of the Packet Forwarding Engines in the router that are actively sending traffic to that output queue. Each output queue is the sum of the total buffers assigned to that output queue across all of the Packet Forwarding Engines in the router. So the output queue itself is virtual, not physical, although the output queue is comprised of physical input queues.

Round-Trip Time Buffering

Although there is no output queue buffering during periods of congestion (no long-term storage), there is a small physical output queue buffer on egress line cards to accommodate the round-trip time for traffic to traverse the fabric from ingress to egress. The round-trip time consists of the time it takes the ingress port to request egress port resources, receive a grant from the egress port for resources, and transmit the data across the fabric.

That means if a packet is not dropped at the router ingress, and the router forwards the packet across the fabric to the egress port, the packet will not be dropped and will be forwarded to the next hop. All packet drops take place in the ingress pipeline.

VOQ Advantages

VOQ architecture provides two major advantages:

- Eliminate Head-of-Line Blocking on page 609
- Increase Fabric Efficiency and Utilization on page 609

Copyright © 2019, Juniper Networks, Inc.

Eliminate Head-of-Line Blocking

VOQ architecture eliminates head-of-line blocking (HOLB) issues. On non-VOQ switches, HOLB occurs when congestion at an egress port affects a different egress port that is not congested. HOLB occurs when the congested port and the non-congested port share the same input queue on an ingress interface.

VOQ architecture avoids HOLB by creating a different dedicated virtual queue for each output queue on each interface.

Because different egress queues do not share the same input queue, a congested egress queue on one port cannot affect an egress queue on a different port. For the same reason, a congested egress queue on one port cannot affect another egress queue on the same port—each output queue has its own dedicated virtual output queue composed of ingress interface input queues.

Performing queue buffering at the ingress interface ensures that the router only sends traffic across the fabric to an egress queue if that egress queue is ready to receive that traffic. If the egress queue is not ready to receive traffic, the traffic remains buffered at the ingress interface.

Increase Fabric Efficiency and Utilization

Traditional output queue architecture has some inherent inefficiencies that VOQ architecture addresses.

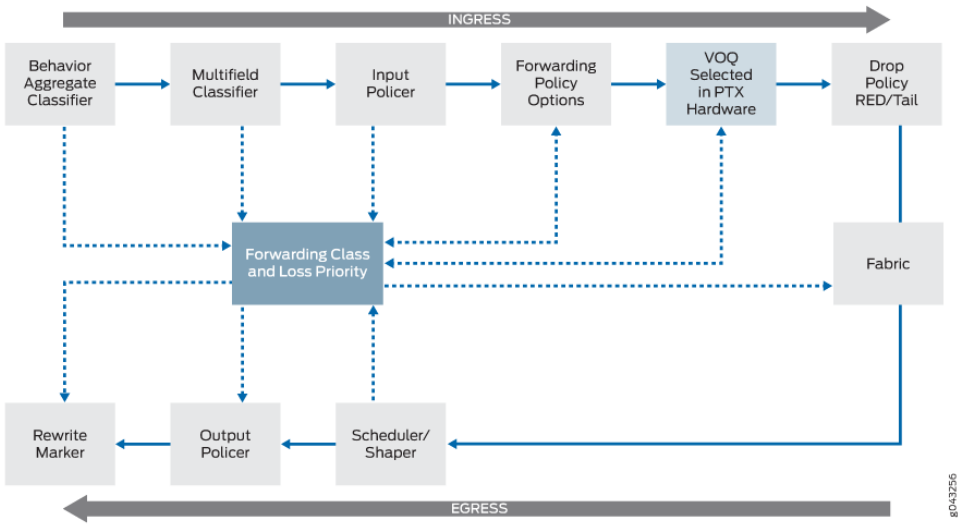
- Packet buffering—Traditional queueing architecture buffers each packet twice in long-term DRAM storage, once at the ingress interface and once at the egress interface. VOQ architecture buffers each packet only once in long-term DRAM storage, at the ingress interface. The fabric is fast enough to be transparent to egress CoS policies, so instead of buffering packets a second time at the egress interface, the router can forward traffic at a rate that does not require deep egress buffers, without affecting the configured egress CoS policies (scheduling).
- Consumption of resources—Traditional queueing architecture sends packets from the ingress interface input queue (buffer), across the fabric, to the egress interface output queue (buffer). At the egress interface, packets might be dropped, even though the router has expended resources transporting the packets across the fabric and storing them in the egress queue. VOQ architecture does not send packets across the fabric to the egress interface until the egress interface is ready to transmit the traffic. This increases system utilization because no resources are wasted transporting and storing packets that are dropped later.

Does VOQ Change How I Configure CoS?

There are no changes to the way you configure the CoS features. Figure 52 on page 610 shows the Juno OS and PTX Series hardware CoS components and VOQ selection, illustrating the sequence in which they interact.

Copyright © 2019, Juniper Networks, Inc.

Figure 52: Packet Flow Through CoS Components on PTX Series Routers



The VOQ selection process is performed by ASICs that use either the behavior aggregate (BA) classifier or the multifield classifier, depending on your configuration, to select one of the eight possible virtual output queues for an egress port. The virtual output queues on the ingress buffer data for the egress port based on your CoS configuration.

Although the CoS features do not change, there are some operational differences with VOQ:

- Random early detection (RED) occurs on the ingress Packet Forwarding Engines. With routers that support only egress output queuing, RED and associated congestion drops occur on the egress. Performing RED on the ingress saves valuable resources and increases router performance.

  Although RED occurs on the ingress with VOQ, there is no change to how you configure the drop profiles.

- Fabric scheduling is controlled through request and grant control messages. Packets are buffered in ingress virtual output queues until the egress Packet Forwarding Engine sends a grant message to the ingress Packet Forwarding Engine indicating it is ready to receive them. For details on fabric scheduling, see "Fabric Scheduling and Virtual Output Queues on PTX Series Routers" on page 613.

See Also

- *Junos OS CoS Components*
- Packet Flow Through the Junos OS CoS Process Overview on page 17
- Understanding How VOQ Works on PTX Series Routers on page 611
- show interfaces voq on page 1468

Copyright © 2019, Juniper Networks, Inc.

Understanding How VOQ Works on PTX Series Routers

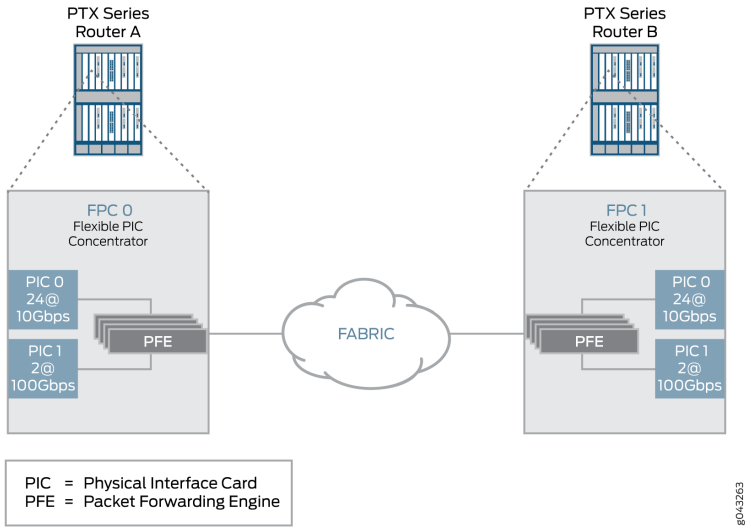
This topic describes how the VOQ process works on PTX Series routers.

- Understanding the Components of the VOQ Process on page 611
- Understanding the VOQ Process on page 612

Understanding the Components of the VOQ Process

Figure 53 on page 611 shows the hardware components of the PTX Series routers involved in the VOQ process.

Figure 53: VOQ Components on PTX Series Routers



These components perform the following functions:

- **Physical Interface Card (PIC)**—Provides the physical connection to various network media types, receiving incoming packets from the network and transmitting outgoing packets to the network.
- **Flexible PIC Concentrator (FPC)**—Connects the PICs installed in it to the other packet transport router components. You can have up to eight FPCs per chassis.
- **Packet Forwarding Engine**—Provides Layer 2 and Layer 3 packet switching and encapsulation and de-encapsulation, forwarding and route lookup functions, and manages packet buffering and the queuing of notifications. The Packet Forwarding Engine receives incoming packets from the PICs installed on the FPC and forwards them through the switch planes to the appropriate destination port.
- **Output queues**—(Not shown) PTX Series routers support up to eight output queues per output port (physical interface). These output queues are controlled by the CoS scheduler configuration, which establishes how to handle the traffic within the output queues for transmission onto the switch fabric. In addition, these egress output queues

Copyright © 2019, Juniper Networks, Inc.

control when packets are sent from the virtual output queues on the ingress to the egress output queues.

Understanding the VOQ Process

PTX Series routers support up to eight output queues per output port (physical interface). These output queues are controlled by the CoS scheduler configuration, which establishes how to handle the traffic within the output queues for transmission onto the fabric. In addition, these egress output queues control when packets are sent from the virtual output queues on the ingress to the egress output queues.

For every egress output queue, the VOQ architecture provides *virtual* queues on each and every ingress Packet Forwarding Engine. These queues are referred to as virtual because the queues physically exist on the ingress Packet Forwarding Engine *only* when the line card actually has packets enqueued to it.

Figure 54 on page 612 shows three ingress Packet Forwarding Engines—PFE0, PFE1, and PFE2. Each ingress Packet Forwarding Engine provides up to eight virtual output queues (PFE*n*.e0.q0 through PFE*n*.e0.q7) for the single egress port 0. The egress Packet Forwarding Engine PFE*n* distributes the bandwidth to each ingress VOQ in a round-robin fashion.

For example, egress PFE N's VOQ e0.q0 has 10 Gbps of bandwidth available to it. PFE 0 has an offered load 10 Gbps to e0.qo, PFE1 and PFE2 have an offered load of 1Gbps to e0.q0. The result is that PFE1 And PFE2 will get 100 percent of their traffic through, while PFE0 will only get 80 percent of its traffic through.

Figure 54: Virtual Output Queues on PTX Series Routers

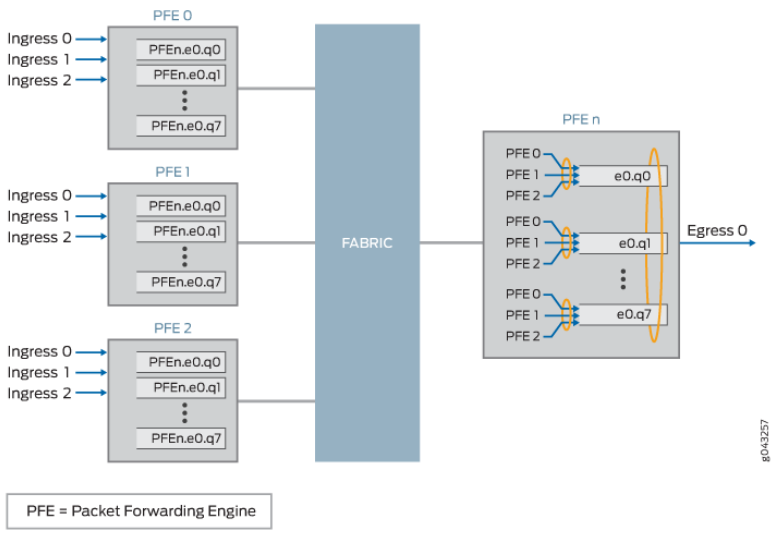
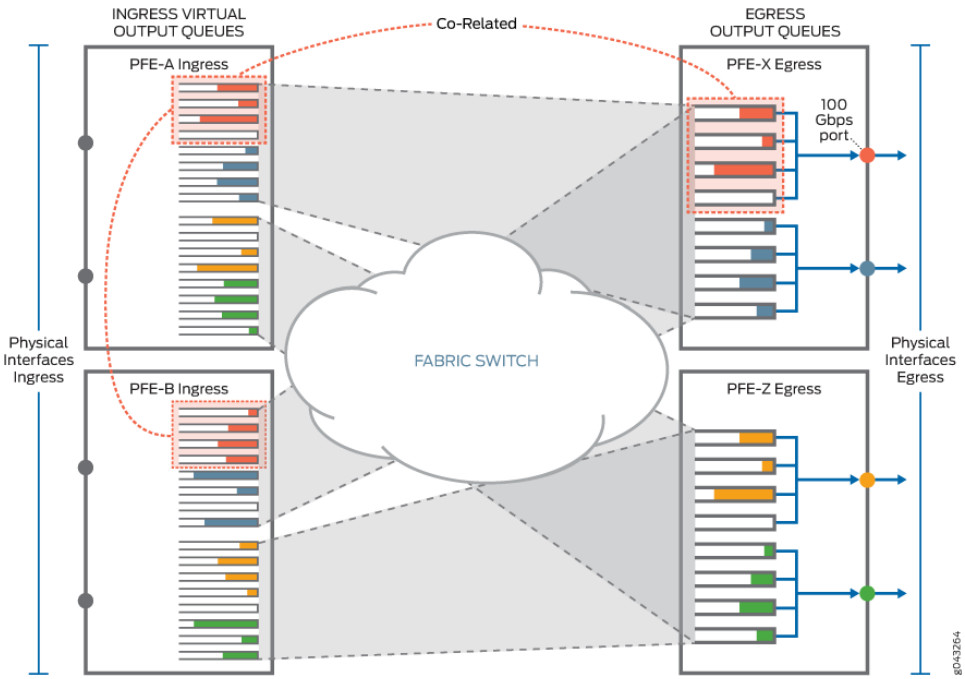


Figure 55 on page 613 illustrates an example of the correlation between the egress output queues and the ingress virtual output queues. On the egress side, PFE-X has a 100 Gbps port, which is configured with four different forwarding classes. As a result, the 100 Gbps egress output port on PFE-X uses four out of eight available egress output queues (as

Copyright © 2019, Juniper Networks, Inc.

denoted by the four queues highlighted with dashed-orange lines on PFE-X), and the VOQ architecture provides four corresponding *virtual* output queues on *each* ingress Packet Forwarding Engine (as denoted by the four virtual queues on PFE-A and PFE-B highlighted with dashed-orange lines). The virtual queues on PFE-A and PFE-B exist only when there is traffic to be sent.

Figure 55: Example of VOQ



See Also

- Introduction to Virtual Output Queues on PTX Series Packet Transport Routers on page 607
- Fabric Scheduling and Virtual Output Queues on PTX Series Routers on page 613
- **show interfaces voq on page 1468**
- *Junos OS CoS Components*

Fabric Scheduling and Virtual Output Queues on PTX Series Routers

This topic describes the fabric scheduling process on PTX Series routers that use VOQ.

VOQ uses request and grant messages to control fabric scheduling on PTX Series routers. The egress Packet Forwarding Engines control data delivery from the ingress virtual output queues by using request and grant messages. The virtual queues buffer packets on the ingress until the egress Packet Forwarding Engine confirms that it is ready to receive them by sending a grant message to the ingress Packet Forwarding Engine.

Copyright © 2019, Juniper Networks, Inc.

Figure 56: Fabric Scheduling and Virtual Output Queues Process

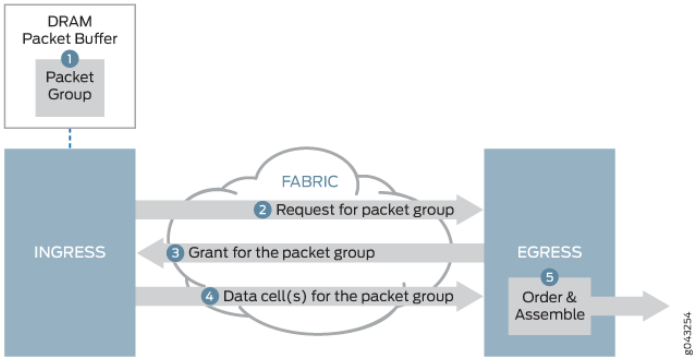


Figure 56 on page 614 illustrates the fabric scheduling process used by PTX Series routers with VOQ. When packets arrive at an ingress port, the ingress pipeline stores the packet in the ingress queue associated with the destination output queue. The router makes the buffering decision after performing the packet lookup. If the packet belongs to a forwarding class for which the maximum traffic threshold has been exceeded, the packet may not be buffered and might be dropped. The scheduling process works as follows:

1. An ingress Packet Forwarding Engine receives a packet and buffers it in virtual queues, then groups the packet with other packets destined for the same egress interface and data output queue.
2. The ingress line card Packet Forwarding Engine sends a request, which contains a reference to the packet group, over the fabric to the egress Packet Forwarding Engine.
3. When there is available egress bandwidth, the egress line card grant scheduler responds by sending a bandwidth grant to the ingress line card Packet Forwarding Engine. .
4. When the ingress line card Packet Forwarding Engine receives the grant from the egress line card Packet Forwarding Engine, the ingress Packet Forwarding Engine segments the packet group and sends all of the pieces over the fabric to the egress Packet Forwarding Engine.
5. The egress Packet Forwarding Engine receives the pieces, reassembles them into the packet group, and enqueues individual packets to a data output queue corresponding to the virtual output queue.

Ingress packets remain in the VOQ on the ingress port input queues until the output queue is ready to accept and forward more traffic.

Under most conditions, the fabric is fast enough to be transparent to egress class-of-service (CoS) policies, so the process of forwarding traffic from the ingress pipeline, across the fabric, to egress ports, does not affect the configured CoS policies for the traffic. The fabric only affects CoS policy if there is a fabric failure or if there is an issue of port fairness.

When a packet ingresses and egresses the same Packet Forwarding Engine (local switching), the packet does not traverse the fabric. However, the router uses the same

Copyright © 2019, Juniper Networks, Inc.

request and grant mechanism to receive egress bandwidth as packets that cross the fabric, so locally routed packets and packets that arrive at a Packet Forwarding Engine after crossing the fabric are treated fairly when the traffic is vying for the same output queue.

See Also

- Introduction to Virtual Output Queues on PTX Series Packet Transport Routers on page 607
- Understanding How VOQ Works on PTX Series Routers on page 611
- Understanding the Packet Forwarding Engine Fairness and Virtual Output Queue Process on page 615
- *Junos OS CoS Components*

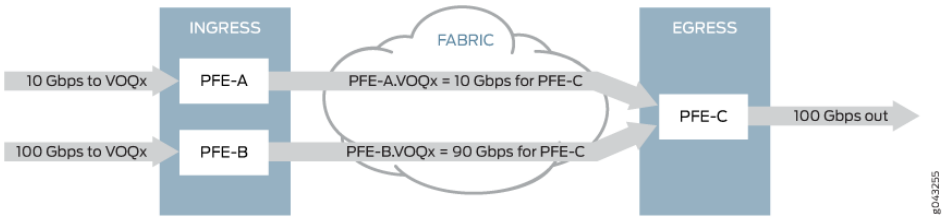
Understanding the Packet Forwarding Engine Fairness and Virtual Output Queue Process

This topic describes the Packet Forwarding Engine fairness scheme used with VOQ on PTX Series routers.

Packet Forwarding Engine fairness means that all Packet Forwarding Engines are treated equally from a egress perspective. If multiple egress Packet Forwarding Engines need to transmit data from the same virtual output queue, they are serviced in round-robin fashion. Servicing of virtual output queues is *not* dependent upon the load that is present at each of the source Packet Forwarding Engines.

Figure 57 on page 615 illustrates the Packet Forwarding Engine fairness scheme used with VOQ in a simple example with three Packet Forwarding Engines. Ingress PFE-A has a single stream of 10 Gbps data destined for VOQ*x* on PFE-C. PFE-B has a single stream of 100 Gbps data also destined for VOQ*x* on PFE-C. On PFE-C, VOQ*x* is serviced by a 100 Gbps interface and that is the only active virtual output queue on that interface.

Figure 57: Packet Forwarding Engine Fairness with Virtual Output Queue Process



In Figure 57 on page 615, we have a total of 110 Gbps of source data destined for a 100 Gbps output interface. As a result, we need to drop 10 Gbps of data. Where does the drop occur and how does this drop affect traffic from PFE-A versus PFE-B?

Because PFE-A and PFE-B are serviced in round-robin fashion by egress PFE-C, all 10 Gbps of traffic from PFE-A makes it through to the egress output port. However, 10 Gbps of data is dropped on PFE-B, allowing only 90 Gbps of data from PFE-B to be sent to PFE-C. So, the 10 Gbps stream has a 0% drop and the 100 Gbps stream has only a 10% drop.

Copyright © 2019, Juniper Networks, Inc.

However, if PFE-A and PFE-B were each sourcing 100 Gbps of data, then they would each drop 50 Gbps of data. This is because the egress PFE-C actually controls the servicing and drain rate on the ingress virtual queues using the round-robin algorithm. With the round-robin algorithm, higher bandwidth sources are always penalized when multiple sources are present. The algorithm attempts to make the two sources equal in bandwidth; however, because it cannot raise the bandwidth of the slower source, it drops the bandwidth of the higher source. The round robin algorithm continues this sequence until the sources have equal egress bandwidth.

Each ingress Packet Forwarding Engine provides up to eight virtual output queues for a single egress port. The egress Packet Forwarding Engine distributes the bandwidth to each ingress virtual output queue; therefore they will receive equal treatment regardless of their presented load. The drain-rate of a queue is the rate at which a queue is draining. The egress Packet Forwarding Engine divides its bandwidth for each output queue equally across the ingress Packet Forwarding Engines. So, the drain-rate of each ingress Packet Forwarding Engine=Drain-rate of output queue/Number of ingress Packet Forwarding Engines.

Handling Congestion

There are two main types of congestion that can occur:

- Ingress congestion — Occurs when the ingress Packet Forwarding Engine has more offered load than the egress can handle. The ingress congestion case, is very similarly to a traditional router in that the queues build-up and once they cross their configured threshold, packets are dropped.
- Egress congestion — Occurs when the sum of all the ingress Packet Forwarding Engines exceeds the capability of the egress router. All drops are performed on the ingress Packet Forwarding Engines. However, the size of the ingress queue is attenuated by the queue's drain-rate (how fast the egress Packet Forwarding Engine is requesting packets). This rate is essentially determined by the rate that requests are being converted in to grants by the egress Packet Forwarding Engine. The egress Packet Forwarding Engine services the request-to-grant conversion in round-robin fashion; it is not dependent on the ingress Packet Forwarding Engines offered load. For instance, if the ingress Packet Forwarding Engine's drain-rate is half of what it expects it to be (as is the case when 2 ingress Packet Forwarding Engines are presenting an oversubscribed load for the target output queue), then the ingress Packet Forwarding Engine's reduce the size of this queue to be half of its original size (when it was getting its full drain rate).

See Also

- Introduction to Virtual Output Queues on PTX Series Packet Transport Routers on page 607
- Understanding How VOQ Works on PTX Series Routers on page 611
- Fabric Scheduling and Virtual Output Queues on PTX Series Routers on page 613
- **show interfaces voq on page 1468**
- *Junos OS CoS Components*

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- show interfaces voq on page 1468
- The Junos OS CoS Components Used to Manage Congestion and Control Service Levels on page 7

Example: Configuring Excess Rate for PTX Series Packet Transport Routers

You can configure excess rate to customize the distribution of available excess bandwidth among the queues for PTX Series Packet Transport Routers. When excess rate is not configured, the excess bandwidth available is distributed in proportion to the transmit rates allocated to the queues.

- Requirements on page 617
- Overview on page 617
- Configuration on page 618
- Verification on page 623

Requirements

This example uses the following hardware and software components:

- One PTX Series Packet Transport Router
- Junos OS Release 12.1X48R2 or later

Overview

This set of examples illustrates how you configure schedulers for the PTX Series Packet Transport Router to distribute the remaining bandwidth (excess rate) among the configured queues.

When you configure excess rate, use the following guidelines:

- The **transmit-rate** statements of the configured schedulers can add up to at most 100 percent.
- All queues on the PTX Series Packet Transport Router have the same excess priority. Excess priority configuration is not supported.
- If a strict-high-priority queue is configured and is rate-limited, this queue gets the rate-limited bandwidth first. Then the configured **transmit-rate** value of other queues is met (regardless of queue priority), and finally the excess bandwidth is distributed in proportion to the configured **excess-rate** values.

BEST PRACTICE: We recommend that you configure rate limit on strict-high queues because the other queues might not meet their guaranteed bandwidths. See transmit-rate.

Copyright © 2019, Juniper Networks, Inc.

Configuration

To configure excess rate, perform one or more of these tasks:

- Configuring Schedulers Without Specifying Excess Rate on page 618
- Configuring Schedulers by Specifying Excess Rate on page 619
- Configuring Schedulers to Control Excess Rate for Non-High-Priority Queues on page 621

Configuring Schedulers Without Specifying Excess Rate

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.

```
set class-of-service schedulers sched_queue_0 transmit-rate percent 20
set class-of-service schedulers sched_queue_1 transmit-rate percent 40
set class-of-service schedulers sched_queue_2 transmit-rate percent 10
set class-of-service schedulers sched_queue_3 transmit-rate percent 10
```

Step-by-Step Procedure

In this example, four queues are configured and each associated scheduler is assigned the indicated transmit rate. Across the four queues, the transmit rate totals to 80 percent. No excess rate is configured. Assuming that each queue has loads greater than or equal to the configured transmit rate, the remaining 20 percent of the bandwidth is distributed in proportion to the configured transmit rates (20:40:10:10):

- sched_queue_0—5% (20% of the guaranteed rate plus 5% of the remaining bandwidth is 25%)
- sched_queue_1—10% (40% of the guaranteed rate plus 10% of the remaining bandwidth is 50%)
- sched_queue_2—2.5% (10% of the guaranteed rate plus 2.5% of the remaining bandwidth is 12.5%)
- sched_queue_3—2.5% (10% of the guaranteed rate plus 2.5% of the remaining bandwidth is 12.5%)

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the schedulers:

1. Create the scheduler for queue 0:

```
[edit class-of-service]
user@host# set schedulers sched_queue_0 transmit-rate percent 20
```

2. Create the scheduler for queue 1:

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service]
user@host# set schedulers sched_queue_1 transmit-rate percent 40
```

3. Create the scheduler for queue 2:

```
[edit class-of-service]
user@host# set schedulers sched_queue_2 transmit-rate percent 10
```

4. Create the scheduler for queue 3:

```
[edit class-of-service]
user@host# set schedulers sched_queue_3 transmit-rate percent 10
```

Results From configuration mode, confirm your configuration by entering the **show class-of-service schedulers** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
sched_queue_0 {
  transmit-rate percent 20;
}
sched_queue_1 {
  transmit-rate percent 40;
}
sched_queue_2 {
  transmit-rate percent 10;
}
sched_queue_3 {
  transmit-rate percent 10;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Configuring Schedulers by Specifying Excess Rate

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.

```
set class-of-service schedulers sched_queue_0 transmit-rate percent 25
set class-of-service schedulers sched_queue_0 excess-rate percent 25
set class-of-service schedulers sched_queue_1 transmit-rate percent 25
set class-of-service schedulers sched_queue_1 excess-rate percent 50
set class-of-service schedulers sched_queue_2 transmit-rate percent 25
set class-of-service schedulers sched_queue_3 transmit-rate percent 25
```

Copyright © 2019, Juniper Networks, Inc.

Step-by-Step Procedure

In this example, four schedulers are configured and each is assigned a transmit rate of 25 percent. Queue 0 is configured with 25 percent and queue 1 with 50 percent of the excess rate. If the offered load through queue 2 is only 10 percent, the remaining bandwidth is distributed as: queue excess rate / total excess rate * remaining bandwidth percentage. If a queue has transmit rate configured but not excess rate, the excess rate for that queue is 1. In this example, the excess rate ratio is 25:50:1:1, which yields the following distribution of the 15 percent remaining bandwidth from queue 2:

- sched_queue_0—4.93% (25 / 76 * 15%)
- sched_queue_1—9.87% (50 / 76 * 15%)
- sched_queue_3—0.197% (1 / 76 * 15%)

When the offered load on queue 2 increases to 25 percent or greater, the other queues get only their configured transmit rates.

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the schedulers:

1. Create the scheduler for queue 0:

```
[edit class-of-service]
user@host# set schedulers sched_queue_0 transmit-rate percent 25
user@host# set schedulers sched_queue_0 excess-rate percent 25
```

2. Create the scheduler for queue 1:

```
[edit class-of-service]
user@host# set schedulers sched_queue_1 transmit-rate percent 25
user@host# set schedulers sched_queue_1 excess-rate percent 50
```

3. Create the scheduler for queue 2:

```
[edit class-of-service]
user@host# set schedulers sched_queue_2 transmit-rate percent 25
```

4. Create the scheduler for queue 3:

```
[edit class-of-service]
user@host# set schedulers sched_queue_3 transmit-rate percent 25
```

Results

From configuration mode, confirm your configuration by entering the **show class-of-service schedulers** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Copyright © 2019, Juniper Networks, Inc.

```
sched_queue_0 {
   transmit-rate percent 25;
   excess-rate percent 25;
}
sched_queue_1 {
   transmit-rate percent 25;
   excess-rate percent 50;
}
sched_queue_2 {
   transmit-rate percent 25;
}
sched_queue_3 {
   transmit-rate percent 25;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Configuring Schedulers to Control Excess Rate for Non-High-Priority Queues

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.

```
set class-of-service schedulers sched_queue_0 transmit-rate percent 90
set class-of-service schedulers sched_queue_0 priority high
set class-of-service schedulers sched_queue_1 transmit-rate percent 10
set class-of-service schedulers sched_queue_1 priority low
set class-of-service schedulers sched_queue_2 excess-rate percent 10
set class-of-service schedulers sched_queue_3 excess-rate percent 30
```

Copyright © 2019, Juniper Networks, Inc.

Step-by-Step Procedure

In this example, the scheduler for queue 0 is configured to transmit up to 90 percent of traffic if there is enough offered load. When the traffic to queue 0 is less than 90 percent, excess rate is configured to distribute the remaining bandwidth in the ratio 1:1:10:30 (when the offered load on queue 1 is greater than 10 percent), which yields the following distribution of the remaining bandwidth from queue 0:

- sched_queue_1—0.0244 * x% (1 / 41 * remaining bandwidth (x)%)
- sched_queue_2—0.244 * x% (10 / 41 * remaining bandwidth (x)%)
- sched_queue_3—0.732 * x% (30 / 41 * remaining bandwidth (x)%)

NOTE: Although the transmit-rate values on queues can add up to at most 100 percent, the excess-rate value does not have this restriction because it is a ratio.

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the schedulers:

1. Create the scheduler for queue 0:

```
[edit class-of-service]
user@host# set schedulers sched_queue_0 transmit-rate percent 90
user@host# set schedulers sched_queue_0 priority high
```

2. Create the scheduler for queue 1:

```
[edit class-of-service]
user@host# set schedulers sched_queue_1 transmit-rate percent 10
user@host# set schedulers sched_queue_1 priority low
```

3. Create the scheduler for queue 2:

```
[edit class-of-service]
user@host# set schedulers sched_queue_2 excess-rate percent 10
```

4. Create the scheduler for queue 3:

```
[edit class-of-service]
user@host# set schedulers sched_queue_3 excess-rate percent 30
```


Copyright © 2019, Juniper Networks, Inc.

Results From configuration mode, confirm your configuration by entering the **show class-of-service schedulers** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
sched_queue_0 {
   transmit-rate percent 90;
   priority high;
}
sched_queue_1 {
   transmit-rate percent 10;
   priority low;
}
sched_queue_2 {
   excess-rate percent 10;
}
sched_queue_3 {
   excess-rate percent 30;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Verifying the Excess Rate Configuration

Purpose Verify that the excess rate configuration is producing the results you expect.

Action From operational mode, enter the **show interfaces queue *interface*** command for the physical interface to verify.

Meaning The show command output lists the traffic by queue and forwarding class names. Verify that the Bytes field for active queues on the specified physical interface match the proportions you expect from the excess rate configuration.

Related Documentation

- How Schedulers Define Output Queue Properties on page 254
- Configuring a Scheduler on page 489
- **excess-rate on page 1107**
- CoS Features and Limitations on PTX Series Routers on page 599

Copyright © 2019, Juniper Networks, Inc.

Identifying the Source of RED Dropped Packets on PTX Series Routers

Copyright © 2019, Juniper Networks, Inc.

This topic describes how to identify the source of random early detection (RED) dropped packets.

Junos OS and PTX Series hardware CoS features use virtual output queues (VOQs) on the ingress to buffer and queue traffic for each egress output queue.

VOQ is a queuing strategy that eliminates congestion drops on the egress and alleviates head-of-line blocking. Head-of-line blocking is a condition in which a queue of packets is blocked from making progress because the packet at the head of the queue is waiting for resources to become available, while other packets behind this packet could be serviced. For example, if the ingress has a single queue for an egress Packet Forwarding Engine, then packets destined for a slow, congested interface can block packets destined for a fast, uncongested interface attached to the same egress Packet Forwarding Engine.

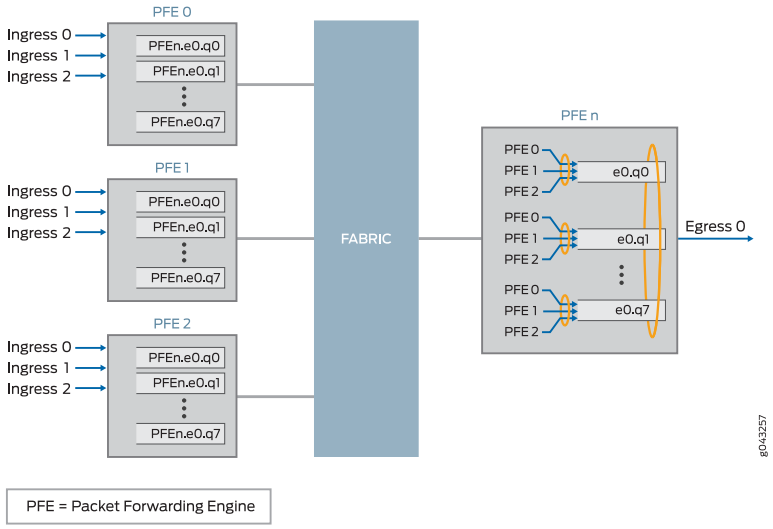
With VOQ, *virtual* queues are maintained on the ingress Packet Forwarding Engines, instead of on the egress Packet Forwarding Engine. However, the scheduling of the ingress virtual output queues is controlled by the egress Packet Forwarding Engine. For every egress output queue (shallow buffer), the VOQ architecture provides *virtual* queues on each and every ingress Packet Forwarding Engine. These queues are referred to as virtual because the queues physically exist on the ingress Packet Forwarding Engine *only* when the line card actually has packets enqueued to it.

Figure 58 on page 625 shows three ingress Packet Forwarding Engines—PFE0, PFE1, and PFE2. Each ingress Packet Forwarding Engine provides up to eight virtual output queues (PFE*n*.e0.q0 through PFE*n*.e0.q7) for the single egress port 0. The egress Packet Forwarding Engine PFE*n* distributes the bandwidth to each ingress VOQ in a round-robin fashion; therefore they will receive equal treatment regardless of their presented load.

For example, egress PFE*n*'s VOQ e0.q0 has 10 Gbps of bandwidth available to it. PFE0 has an offered load of 10 Gbps to e0.qo, whereas PFE1 and PFE2 have an offered load of 1Gbps to e0.q0. The result is that PFE1 and PFE2 get 100 percent of their traffic through, wheras PFE0 gets only 80 percent of its traffic through.

Figure 58: Virtual Output Queuing on PTX Series Routers

Copyright © 2019, Juniper Networks, Inc.



When congestion occurs because of the load on the egress output queue, the ingress VOQs corresponding to the egress output queue contain RED dropped packets.

NOTE: For more information about VOQ, see "Understanding Virtual Output Queues on PTX Series Packet Transport Routers" on page 607.

Using the following procedure, you can identify the ingress Packet Forward Engine (in terms of ingress traffic) that is contributing to the egress congestion.

To determine which ingress Packet Forwarding Engine is contributing to the RED dropped packets:

1. Determine whether there are RED dropped packets on the egress link.
   a. Run the **show interfaces queue *interface-name*** command on the egress interface.

      ```
      user@host> show interfaces queue et-7/0/0
      ```

   b. In the **show** output, determine whether the interface is experiencing RED dropped packets, by locating the RED-dropped packets field and checking whether its value is greater than zero.

      The following example shows RED-dropped statistics for the egress Ethernet interface configured on port 0 of PIC 0, located on the FPC in slot 7.

```
user@host> show interfaces queue et-7/0/0
Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 206, SNMP ifIndex: 790
Forwarding classes: 16 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: fc0
```


Copyright © 2019, Juniper Networks, Inc.

```
  Queued:
    Packets              :            539433200             14896082 pps
    Bytes                :          38302319880           8461137824 bps
  Transmitted:
    Packets              :             67108815              1859497 pps
    Bytes                :           4294964160            952062464 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :            472324385             13036585 pps
    RED-dropped bytes    :          34007355720           7509075360 bps
Queue: 1, Forwarding classes: fc1
  Queued:
    Packets              :            539433555             14877096 pps
    Bytes                :          38302345472           8450201072 bps
  Transmitted:
    Packets              :             67108811              1859498 pps
    Bytes                :           4294963904            952062976 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :            472324744             13017598 pps
    RED-dropped bytes    :          34007381568           7498138096 bps
Queue: 2, Forwarding classes: fc2
  Queued:
    Packets              :            539433811             14892745 pps
    Bytes                :          38302363728           8459214984 bps
  Transmitted:
    Packets              :             67108833              1859501 pps
    Bytes                :           4294965312            952064512 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :            472324978             13033244 pps
    RED-dropped bytes    :          34007398416           7507150472 bps
Queue: 3, Forwarding classes: fc3
  Queued:
    Packets              :            539433461             14879323 pps
    Bytes                :          38302338584           8451484208 bps
  Transmitted:
    Packets              :             67108826              1859498 pps
    Bytes                :           4294964864            952062976 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :            472324635             13019825 pps
    RED-dropped bytes    :          34007373720           7499421232 bps
Queue: 4, Forwarding classes: fc4
  Queued:
    Packets              :            539433755             14884190 pps
    Bytes                :          38302359616           8454286816 bps
  Transmitted:
    Packets              :             67108843              1859508 pps
    Bytes                :           4294965952            952068096 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :            472324912             13024682 pps
    RED-dropped bytes    :          34007393664           7502218720 bps
Queue: 5, Forwarding classes: fc5
```

```
  Queued:
    Packets              :            539433849            14892950 pps
    Bytes                :          38302366384          8459333176 bps
  Transmitted:
    Packets              :             67108843             1859497 pps
    Bytes                :           4294965952           952062464 bps
    Tail-dropped packets :                    0                   0 pps
    RL-dropped packets   :                    0                   0 pps
    RL-dropped bytes     :                    0                   0 bps
    RED-dropped packets  :            472325006            13033453 pps
    RED-dropped bytes    :          34007400432          7507270712 bps
Queue: 6, Forwarding classes: fc6
  Queued:
    Packets              :            539434160            14879808 pps
    Bytes                :          38302388632          8451762856 bps
  Transmitted:
    Packets              :             67108861             1859514 pps
    Bytes                :           4294967104           952071168 bps
    Tail-dropped packets :                    0                   0 pps
    RL-dropped packets   :                    0                   0 pps
    RL-dropped bytes     :                    0                   0 bps
    RED-dropped packets  :            472325299            13020294 pps
    RED-dropped bytes    :          34007421528          7499691688 bps
Queue: 7, Forwarding classes: fc7
  Queued:
    Packets              :            539434364            14900946 pps
    Bytes                :          38302403328          8463940000 bps
  Transmitted:
    Packets              :             67108860             1859496 pps
    Bytes                :           4294967040           952061952 bps
    Tail-dropped packets :                    0                   0 pps
    RL-dropped packets   :                    0                   0 pps
    RL-dropped bytes     :                    0                   0 bps
    RED-dropped packets  :            472325504            13041450 pps
    RED-dropped bytes    :          34007436288          7511878048 bps
```

2. If the interface is experiencing RED dropped packets, run the **show interface voq** ***interface-name*** command on the egress interface that is experiencing the RED dropped packets.

```
user@host> show interfaces voq et-7/0/0 non-zero
```

TIP: When using the **show interfaces voq** command, you can use command filters to help locate the exact queue. For command usage, see show interfaces voq.

3. In the **show** output, determine whether the interface is experiencing RED dropped packets.

   The following example shows the count of the ingress RED-dropped packets for the egress Ethernet interface configured on port 0 of PIC 0, located on the FPC in slot 7.

Copyright © 2019, Juniper Networks, Inc.

The sample output shows that the cause of the congestion is the ingress Packet Forwarding Engine PFE 0, on FPC number 4, and the ingress Packet Forwarding Engine PFE 0 on FPC number 6, as denoted by the count of RED-dropped packets.

```
user@host> show interfaces voq et-7/0/0 non-zero
Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 156, SNMP ifIndex: 699

Queue: 0, Forwarding classes: q00

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              6834995                96929 pps
      RED-dropped bytes    :           5249276160             595537368 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              6835203                96964 pps
      RED-dropped bytes    :           5249435904             595749256 bps

Queue: 1, Forwarding classes: q01

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              6834998                96967 pps
      RED-dropped bytes    :           5249278464             595766280 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              6835201                96627 pps
      RED-dropped bytes    :           5249434368             593677664 bps

Queue: 2, Forwarding classes: q02

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              6834997                96921 pps
      RED-dropped bytes    :           5249277696             595482712 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              6835205                96827 pps
      RED-dropped bytes    :           5249437440             594907344 bps

Queue: 3, Forwarding classes: q03

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              6834997                96961 pps
      RED-dropped bytes    :           5249277696             595731736 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              6835202                96522 pps
      RED-dropped bytes    :           5249435136             593031808 bps

Queue: 4, Forwarding classes: q04

  FPC number: 4
```

```
    PFE: 0
      RED-dropped packets  :              6834995                97021 pps
      RED-dropped bytes    :           5249276160             596099296 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              6835199                96935 pps
      RED-dropped bytes    :           5249432832             595572304 bps

Queue: 5, Forwarding classes: q05

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              6834996                96949 pps
      RED-dropped bytes    :           5249276928             595656872 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              6835204                96899 pps
      RED-dropped bytes    :           5249436672             595348960 bps

Queue: 6, Forwarding classes: q06

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              6835000                97019 pps
      RED-dropped bytes    :           5249280000             596088832 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              6835201                96916 pps
      RED-dropped bytes    :           5249434368             595455624 bps

Queue: 7, Forwarding classes: q07

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              6834999                96929 pps
      RED-dropped bytes    :           5249279232             595536704 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              6835202                96941 pps
      RED-dropped bytes    :           5249435136             595609968 bps
```

NOTE: For an aggregate interface, follow the same steps, but you must run the **show interface queue** command on each child link of the aggregate interface to determine which child egress link is experiencing the congestion. Then run the **show interface voq** command on that child link to determine which ingress Packet Forward Engine is contributing to the congestion.

Related Documentation

- Understanding Virtual Output Queues on PTX Series Packet Transport Routers on page 607
- show interfaces voq on page 1468

Copyright © 2019, Juniper Networks, Inc.

- show interfaces queue on page 1421

Configuring Virtual LAN Queuing and Shaping on PTX Series Routers

You can enable virtual LAN (VLAN) queuing on 100-Gigabit Ethernet interfaces on PTX Series Packet Transport Routers and specify a traffic-shaping rate for each VLAN. In conjunction, you can also configure other class-of-service (CoS) features, including classifiers, schedulers, and rewrite rules.

- Only 100-Gigabit Ethernet interfaces are supported.
- You can configure a maximum of 10 VLANs on each interface.
- The maximum shaping rate cannot exceed 100 Gbps for all VLANs configured on an interface.
- Aggregated Ethernet interfaces are not supported.
- The interface wildcard character (*) is not supported—for example, the following configuration is not supported and should not be used:

```
set class-of-service interface et-* unit * shaping-rate 1g
set class-of-service interface et-* unit * scheduler-map sch0
```

To configure per-VLAN queuing and traffic shaping on PTX Series routers:

1. Enable the reception and transmission of 8021.q VLAN-tagged frames on the interface:

```
[edit interfaces et-fpc/pic/port]
user@host# set vlan-tagging
```

2. Configure logical interface properties.

   a. Specify a VLAN identifier for each logical interface:

```
[edit interfaces et-fpc/pic/port unit logical-unit-number]
user@host# set vlan-id number
```

NOTE: You can specify a maximum of 10 VLAN identifiers for each physical interface.

   b. Specify a protocol family and IP address for each logical interface:

```
[edit interfaces et-fpc/pic/port unit logical-unit-number]
user@host# set family (inet | inet6 | mpls) address ip-address
```

3. Enable per-VLAN queuing on the interface:

Copyright © 2019, Juniper Networks, Inc.

```
[edit interfaces et-fpc/pic/port]
user@host# set per-unit-scheduler
```

4. Configure per-VLAN traffic shaping by specifying the amount of bandwidth to be allocated to each logical interface:

```
[edit class-of-service interfaces et-fpc/pic/port unit logical-unit-number]
user@host# set shaping-rate rate
```

NOTE: The shaping rate for all VLANs cannot exceed 100 percent of the bandwidth available on the interface (100 Gbps).

5. (Optional) Configure one or more classifiers and apply them to the logical interface.
   a. Define one or more behavior aggregate (BA) classifiers:

```
[edit class-of-service classifiers]
user@host# set classifier-type classifier-name
```

   b. Define one or more forwarding classes for each classifier:

```
[edit class-of-service classifiers classifier-type classifier-name]
user@host# set forwarding-class forwarding-class-name loss-priority level
   code-points [ aliases ] [ bit-patterns ]
```

   c. Apply one or more classifiers to the logical interface:

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

6. (Optional) Configure one or more rewrite rules to set CoS bits on outgoing packets.
   a. Define one or more rewrite rules:

```
[edit class-of-service rewrite-rules]
user@host# set traffic-type rewrite-rule-name
```

   b. Define one or more forwarding classes for each rewrite rule:

```
[edit class-of-service rewrite-rules traffic-type rewrite-rule-name]
user@host# set forwarding-class forwarding-class-name loss-priority level
   code-points [ aliases ] [ bit-patterns ]
```

   c. Apply one or more rewrite rules to the logical interface for outgoing traffic:

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
user@host# set rewrite-rules traffic-type rewrite-rule-name
```

7. (Optional) Configure one or more scheduler maps and apply them to the logical interface, that is, VLAN. Use a scheduler map to associate the properties of the output queues you define in schedulers with forwarding classes.

   a. Specify the name of a scheduler map:

   ```
   [edit class-of-service]
   user@host# set scheduler-maps scheduler-map-name
   ```

   b. Specify the name of a forwarding class to associate with the scheduler map:

   ```
   [edit class-of-service scheduler-maps scheduler-map-name]
   user@host set forwarding-class forwarding-class-name
   ```

   c. Specify the name of a scheduler configured at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level to associate with the scheduler map:

   ```
   [edit class-of-service scheduler-maps scheduler-map-name]
   user@host# set schedulers scheduler-name
   ```

   d. Apply the scheduler map to the logical interface, that is, VLAN:

   ```
   [edit class-of-service]
   user@host# set interfaces et-fpc/pic/port unit logical-unit-number scheduler-map
     scheduler-map-name
   ```

Related Documentation

- The Junos OS CoS Components Used to Manage Congestion and Control Service Levels on page 7
- Example: Configuring Virtual LAN Queuing and Shaping in PTX Series Packet Transport Routers on page 634
- per-unit-scheduler on page 1226
- shaping-rate on page 1265

Copyright © 2019, Juniper Networks, Inc.

Example: Configuring Virtual LAN Queuing and Shaping in PTX Series Packet Transport Routers

You can enable virtual LAN (VLAN) queuing on 100-Gigabit Ethernet interfaces on PTX Series Packet Transport Routers and specify a traffic-shaping rate for each VLAN.

- Requirements on page 634
- Overview on page 634
- Configuration on page 634

Requirements

This example uses the following hardware and software components:

- Junos OS Release 13.2 or later.
- One PTX5000 router.

Overview

This example shows how to configure 10 VLANs, enable class-of-service (CoS) queuing, and specify a traffic-shaping rate for each VLAN. The total traffic-shaping rate for all VLANs cannot exceed 100 Gbps.

Configuration

CLI Quick Configuration

To configure VLAN queuing and traffic shaping on the PTX5000 router, copy the following commands and paste them into the terminal window of the router:

```
[edit]
set interfaces et-5/0/0 vlan-tagging
set interfaces et-5/0/0 per-unit-scheduler
set interfaces et-5/0/0 unit 0 vlan-id 0
set interfaces et-5/0/0 unit 1 vlan-id 1
set interfaces et-5/0/0 unit 2 vlan-id 2
set interfaces et-5/0/0 unit 3 vlan-id 3
set interfaces et-5/0/0 unit 4 vlan-id 4
set interfaces et-5/0/0 unit 5 vlan-id 5
set interfaces et-5/0/0 unit 6 vlan-id 6
set interfaces et-5/0/0 unit 7 vlan-id 7
set interfaces et–5/0/0 unit 8 vlan-id 8
set interfaces et-5/0/0 unit 9 vlan-id 9
set class-of-service interfaces et-5/0/0 unit 0 shaping-rate 5g
set class-of-service interfaces et-5/0/0 unit 1 shaping-rate 10g
set class-of-service interfaces et-5/0/0 unit 2 shaping-rate 20g
set class-of-service interfaces et-5/0/0 unit 3 shaping-rate 5g
set class-of-service interfaces et-5/0/0 unit 4 shaping-rate 10g
set class-of-service interfaces et-5/0/0 unit 5 shaping-rate 10g
set class-of-service interfaces et-5/0/0 unit 6 shaping-rate 5g
set class-of-service interfaces et-5/0/0 unit 7 shaping-rate 5g
set class-of-service interfaces et-5/0/0 unit 8 shaping-rate 10g
set class-of-service interfaces et-5/0/0 unit 9 shaping-rate 20g
```


Copyright © 2019, Juniper Networks, Inc.

Step-by-Step Procedure

To configure the PTX5000 router:

1. Enable the reception and transmission of 8021.q VLAN-tagged frames on the interface:

```
[edit interfaces]
user@host# set et-5/0/0 vlan-tagging
```

2. Specify a VLAN identifier for each logical interface:

```
[edit interfaces]
user@host# set et-5/0/0 unit 0 vlan-id 0
user@host# set et-5/0/0 unit 1 vlan-id 1
user@host# set et-5/0/0 unit 2 vlan-id 2
user@host# set et-5/0/0 unit 3 vlan-id 3
user@host# set et-5/0/0 unit 4 vlan-id 4
user@host# set et–5/0/0 unit 5 vlan-id 5
user@host# set et-5/0/0 unit 6 vlan-id 6
user@host# set et-5/0/0 unit 7 vlan-id 7
user@host# set et-5/0/0 unit 8 vlan-id 8
user@host# set et-5/0/0 unit 9 vlan-id 9
```

3. Configure per-VLAN traffic shaping by specifying the amount of bandwidth to be allocated to each logical interface:

```
[edit class-of-service interfaces]
user@host# set et-5/0/0 unit 0 shaping-rate 5g
user@host# set et-5/0/0 unit 1 shaping-rate 10g
user@host# set et-5/0/0 unit 2 shaping-rate 20g
user@host# set et-5/0/0 unit 3 shaping-rate 5g
user@host# set et-5/0/0 unit 4 shaping-rate 10g
user@host# set et-5/0/0 unit 5 shaping-rate 10g
user@host# set et-5/0/0 unit 6 shaping-rate 5 g
user@host# set et-5/0/0 unit 7 shaping-rate 5g
user@host# set et-5/0/0 unit 8 shaping-rate 10g
user@host# set et-5/0/0 unit 0 shaping-rate 20g
```

Results

Confirm your results by entering the **show interfaces** and **show class-of-service** commands:

```
user@host# show interfaces
  et-5/0/0  {
      vlan-tagging;
      unit 0 {
          vlan-id 0;
      }
      unit 1 {
          vlan-id 1;
      }
      unit 2 {
```

Copyright © 2019, Juniper Networks, Inc.

```
            vlan-id 2;
        }
        unit 3 {
            vlan-id 3;
        }
        unit 4 {
            vlan-id 4;
        }
        unit 5 {
            vlan-id 5;
        }
        unit 6 {
            vlan-id 6;
        }
        unit 7 {
            vlan-id 7;
        }
        unit 8 {
            vlan-id 8;
        }
        unit 9 {
            vlan-id 9;
        }
    }
```

```
user@host# show class-of-service
  interfaces {
      et-5/0/3 {
          unit 0 {
              shaping-rate 5g;
          }
          unit 1 {
              shaping-rate 10g;
          }
          unit 2 {
              shaping-rate 20g;
          }
          unit 3 {
              shaping-rate 5g;
          }
          unit 4 {
              shaping-rate 10g;
          }
          unit 5 {
              shaping-rate 10g;
          }
          unit 6 {
              shaping-rate 5g;
          }
          unit 7 {
              shaping-rate 5g;
          }
          unit 8 {
              shaping-rate 10g;
          }
          unit 9 {
              shaping-rate 20g;
          }
      }
  }
```


Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring Virtual LAN Queuing and Shaping on PTX Series Routers on page 631
- **per-unit-scheduler on page 1226**
- **shaping-rate on page 1265**

Example: Configuring Strict-Priority Scheduling on a PTX Series Router

This example shows how to configure strict-priority scheduling for a physical interface on a PTX Series router.

- Requirements on page 637
- Overview on page 637
- Configuration on page 638
- Verification on page 642

Requirements

This example uses the following hardware and software components:

- One PTX Series Packet Transport Router
- One or more routers that provide input packets and receive output packets
- Junos OS Release 13.3 or later

Overview

This example illustrates how you configure strict-priority scheduling for a physical interface on a PTX Series router to perform processing of queues in strict-priority order. Queues in the guaranteed region with the same priority are processed in round-robin fashion. Queues in the excess region are processed based on the WRR algorithm.

When you configure strict-priority scheduling, use the following guidelines:

- The configured **transmit-rate** does not affect the queue drain rate because packets are processed in order of queue priority.
- You can configure only one queue with **strict-high** priority at the **[edit class-of-service schedulers *scheduler-name* priority]** hierarchy level.
- You cannot configure both **transmit-rate exact** and **strict-high** priority at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level.
- You cannot configure **scheduler-map** or **shaping-rate** on an interface where you configure an output traffic control profile.
- You cannot configure **transmit-rate** on a queue with **low** priority or the commit will fail.

Copyright © 2019, Juniper Networks, Inc.

NOTE: If a strict-high priority queue is constantly loaded to 100 percent of traffic capacity, other queues are starved. Queue starvation can cause the interface hardware to generate critical interrupts.

In Figure 59 on page 638, the PTX Series router has inputs from Router A, et-1/1/15 and et-1/1/12, and an output to Router B, et-7/1/12. This example configures classification on the two ingress Interfaces and configures strict-priority scheduling on the egress interface.

Figure 59: Topology for Configuring Strict-Priority Scheduling on a PTX Series Router

Configuration

Configuring Strict-Priority Scheduling

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.

```
set class-of-service classifiers dscp cls forwarding-class be loss-priority high code-points
  000000
set class-of-service classifiers dscp cls forwarding-class ef loss-priority medium-high
  code-points 000001
set class-of-service classifiers dscp cls forwarding-class af loss-priority medium-low
  code-points 000010
set class-of-service classifiers dscp cls forwarding-class nc loss-priority low code-points
  000011
set class-of-service classifiers dscp cls forwarding-class af11 loss-priority low code-points
  000100
set class-of-service classifiers dscp cls forwarding-class af12 loss-priority low code-points
  000101
set class-of-service classifiers dscp cls forwarding-class af13 loss-priority low code-points
  000110
set class-of-service classifiers dscp cls forwarding-class nc2 loss-priority low code-points
  000111
set class-of-service forwarding-classes queue 0 be
set class-of-service forwarding-classes queue 1 ef
set class-of-service forwarding-classes queue 2 af
set class-of-service forwarding-classes queue 3 nc
set class-of-service forwarding-classes queue 4 af11
set class-of-service forwarding-classes queue 5 af12
set class-of-service forwarding-classes queue 6 af13
set class-of-service forwarding-classes queue 7 nc2
set class-of-service traffic-control-profiles tcp1 scheduler-map sch0
set class-of-service traffic-control-profiles tcp1 strict-priority-scheduler
set class-of-service interfaces et-1/1/12 unit 0 classifiers dscp cls
```


Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service interfaces et-1/1/15 unit 0 classifiers dscp cls
set class-of-service interfaces et-7/1/12 output-traffic-control-profile tcp1
set class-of-service scheduler-maps sch0 forwarding-class be scheduler be_sch
set class-of-service scheduler-maps sch0 forwarding-class ef scheduler ef_sch
set class-of-service scheduler-maps sch0 forwarding-class af scheduler af_sch
set class-of-service scheduler-maps sch0 forwarding-class nc scheduler nc_sch
set class-of-service scheduler-maps sch0 forwarding-class af11 scheduler af11_sch
set class-of-service scheduler-maps sch0 forwarding-class af12 scheduler af12_sch
set class-of-service scheduler-maps sch0 forwarding-class af13 scheduler af13_sch
set class-of-service scheduler-maps sch0 forwarding-class nc2 scheduler nc2_sch
set class-of-service schedulers be_sch transmit-rate percent 60
set class-of-service schedulers be_sch priority high
set class-of-service schedulers ef_sch transmit-rate percent 5
set class-of-service schedulers ef_sch priority medium-high
set class-of-service schedulers af_sch transmit-rate percent 5
set class-of-service schedulers af_sch priority high
set class-of-service schedulers nc_sch transmit-rate percent 5
set class-of-service schedulers nc_sch priority strict-high
set class-of-service schedulers af11_sch transmit-rate percent 5
set class-of-service schedulers af11_sch priority high
set class-of-service schedulers af12_sch transmit-rate percent 5
set class-of-service schedulers af12_sch priority medium-high
set class-of-service schedulers af13_sch transmit-rate percent 5
set class-of-service schedulers af13_sch priority medium-low
set class-of-service schedulers nc2_sch priority low
```

Step-by-Step Procedure

In this example, eight schedulers are configured based on eight DSCP classifier configurations. Each associated scheduler is assigned a priority and transmit rate, although the transmit rate is ignored by the strict-priority scheduler. The scheduler map sch0 is configured with the mapping of forwarding classes to schedulers. Within the traffic control profile tcp1, the scheduler map and the strict-priority scheduler feature are configured. Two input interfaces on the PTX Series router, et-1/1/12 and et-1/1/15, are configured with the DSCP classifiers. The output traffic control profile on et-7/1/12 is configured with the traffic control profile tcp1.

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure strict-priority scheduling:

1. Configure the DSCP forwarding classes.

```
[edit class-of-service dscp cls]
user@host# set forwarding-class be loss-priority high code-points 000000
user@host# set forwarding-class ef loss-priority medium-high code-points 000001
user@host# set forwarding-class af loss-priority medium-low code-points 000010
user@host# set forwarding-class nc loss-priority low code-points 000011
user@host# set forwarding-class af11 loss-priority low code-points 000100
user@host# set forwarding-class af12 loss-priority low code-points 000101
user@host# set forwarding-class af13 loss-priority low code-points 000110
user@host# set forwarding-class nc2 loss-priority low code-points 000111
```

Copyright © 2019, Juniper Networks, Inc.

2. Configure the mapping of queues to forwarding classes.

```
[edit class-of-service forwarding-classes]
user@host# set queue 0 be
user@host# set queue 1 ef
user@host# set queue 2 af
user@host# set queue 3 nc
user@host# set queue 4 af11
user@host# set queue 5 af12
user@host# set queue 6 af13
user@host# set queue 7 nc2
```

3. Configure the transmit rate and priority for each scheduler.

   Although you can configure a transmit rate, the value that you configure is overridden by the strict-priority scheduler.

```
[edit class-of-service]
user@host# set schedulers be_sch transmit-rate percent 60
user@host# set schedulers be_sch priority high
user@host# set schedulers ef_sch transmit-rate percent 5
user@host# set schedulers ef_sch priority medium-high
user@host# set schedulers af_sch transmit-rate percent 5
user@host# set schedulers af_sch priority high
user@host# set schedulers nc_sch transmit-rate percent 5
user@host# set schedulers nc_sch priority strict-high
user@host# set schedulers af11_sch transmit-rate percent 5
user@host# set schedulers af11_sch priority high
user@host# set schedulers af12_sch transmit-rate percent 5
user@host# set schedulers af12_sch priority medium-high
user@host# set schedulers af13_sch transmit-rate percent 5
user@host# set schedulers af13_sch priority medium-low
user@host# set schedulers nc2_sch priority low
```

4. Configure the scheduler map with the mapping of forwarding classes to schedulers.

```
[edit class-of-service scheduler-maps sch0]
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 af11 scheduler af11_sch
user@host# set forwarding-class af12 scheduler af12_sch
user@host# set forwarding-class af13 scheduler af13_sch
user@host# set forwarding-class nc2 scheduler nc2_sch
```

5. Configure the traffic control profile to do strict-priority scheduling and define the scheduler map to use.

```
[edit class-of-service traffic-control-profiles tcp1]
user@host# set scheduler-map sch0
user@host# set strict-priority-scheduler
```


Copyright © 2019, Juniper Networks, Inc.

6. Apply the classifiers to the input interfaces, and the traffic control profile to the output interface.

```
[edit class-of-service interfaces]
user@host# set et-1/1/12 unit 0 classifiers dscp cls
user@host# set et-1/1/15 unit 0 classifiers dscp cls
user@host# set et-7/1/12 output-traffic-control-profile tcp1
```

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

```
[edit]
user@host# show class-of-service
forwarding-classes {
   queue 0 be;
   queue 1 ef;
   queue 2 af;
   queue 3 nc;
   queue 4 af11;
   queue 5 af12;
   queue 6 af13;
   queue 7 nc2;
}
interfaces {
   et-1/1/12 {
      unit 0 {
         classifiers {
            dscp cls;
         }
      }
   }
   et-1/1/15 {
      unit 0 {
         classifiers {
            dscp cls;
         }
      }
   }
   et-7/1/12 {
      output-traffic-control-profile tcp1;
   }
}
scheduler-maps {
   sch0 {
      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 af11 scheduler af11_sch;
      forwarding-class af12 scheduler af12_sch;
      forwarding-class af13 scheduler af13_sch;
```

Copyright © 2019, Juniper Networks, Inc.

```
      forwarding-class nc2 scheduler nc2_sch;
    }
  }
  schedulers {
    be_sch {
      transmit-rate percent 60;
      priority high;
    }
    ef_sch {
      transmit-rate percent 5;
      priority medium-high;
    }
    af_sch {
      transmit-rate percent 5;
      priority high;
    }
    nc_sch {
      transmit-rate percent 5;
      priority strict-high;
    }
    af11_sch {
      transmit-rate percent 5;
      priority high;
    }
    af12_sch {
      transmit-rate percent 5;
      priority medium-high;
    }
    af13_sch {
      transmit-rate percent 5;
      priority medium-low;
    }
    nc2_sch {
      priority low;
    }
  }
  traffic-control-profiles {
    tcp1 {
      scheduler-map sch0;
      strict-priority-scheduler;
    }
  }
```

Verification

Verifying Strict-Priority Scheduling

Purpose Verify that the strict-priority scheduling configuration is producing the results you expect.

Action From operational mode, enter the **show interfaces queue interface-name *interface-name*** command and select the output physical interface to verify.

```
user@host> show interfaces queue interface-name et-7/1/12
```


Copyright © 2019, Juniper Networks, Inc.

```
Physical interface: et-7/1/12, Enabled, Physical link is Up
  Interface index: 231, SNMP ifIndex: 612
Forwarding classes: 16 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: be
  Queued:
    Packets              :               394488               131507 pps
    Bytes                :            591732000            1578084848 bps
  Transmitted:
    Packets              :               394488               131507 pps
    Bytes                :            591732000            1578084848 bps
    Tail-dropped packets :                    0                    0 pps
    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: ef
  Queued:
    Packets              :               234498                82115 pps
    Bytes                :            352963584             988886784 bps
  Transmitted:
    Packets              :                82425                27551 pps
    Bytes                :            123637500             330618176 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :               152073                54564 pps
    RED-dropped bytes    :            229326084             658268608 bps
Queue: 2, Forwarding classes: af
  Queued:
    Packets              :               345175               115068 pps
    Bytes                :            517762500            1380824240 bps
  Transmitted:
    Packets              :               345175               115068 pps
    Bytes                :            517762500            1380824240 bps
    Tail-dropped packets :                    0                    0 pps
    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: nc
  Queued:
    Packets              :               986224               328769 pps
    Bytes                :           1479336000            3945236360 bps
  Transmitted:
    Packets              :               986224               328769 pps
    Bytes                :           1479336000            3945236360 bps
    Tail-dropped packets :                    0                    0 pps
    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: 4, Forwarding classes: af11
  Queued:
    Packets              :               493110               164383 pps
    Bytes                :            739665000            1972606056 bps
  Transmitted:
    Packets              :               493110               164383 pps
    Bytes                :            739665000            1972606056 bps
```

```
    Tail-dropped packets :                    0                    0 pps
    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: 5, Forwarding classes: af12
  Queued:
    Packets              :               461830               164375 pps
    Bytes                :            695777416            1981272728 bps
  Transmitted:
    Packets              :                82778                27543 pps
    Bytes                :            124167000             330521208 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :               379052               136832 pps
    RED-dropped bytes    :            571610416            1650751520 bps
Queue: 6, Forwarding classes: af13
  Queued:
    Packets              :               462258               164556 pps
    Bytes                :            696421280            1983445256 bps
  Transmitted:
    Packets              :                82973                27637 pps
    Bytes                :            124459500             331648480 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :               379285               136919 pps
    RED-dropped bytes    :            571961780            1651796776 bps
Queue: 7, Forwarding classes: nc2
  Queued:
    Packets              :               227750                82215 pps
    Bytes                :            343447000             991843712 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :               227750                82215 pps
    RED-dropped bytes    :            343447000             991843712 bps
```

Meaning The **show** command output lists the traffic by queue and forwarding class names. The Bytes field under the Transmitted field for each queue shows the actual bytes transmitted.

From the sample output, you can see that the strict-high queue gets the highest priority and transmits without drops. The high-priority queues are then transmitted. The medium-high and medium-low priority queues are processed in a round-robin fashion. The low-priority queue is starved.

Keep in mind the following conditions that apply to strict-priority scheduling:

Copyright © 2019, Juniper Networks, Inc.

- If the traffic on the output interface is undersubscribed, no queue should show dropped traffic.
- The strict-high queue is processed first, followed by the high-priority queues (in a round-robin fashion), and finally all remaining queues in the guaranteed region (in a round-robin fashion).
- If the ingress traffic exceeds the capacity of the output interface, the queues are processed in strict-priority order.
- Queues in the excess region are processed based on the WRR algorithm.

Related Documentation

- Understanding Scheduling on PTX Series Routers on page 604
- Example: Configuring Excess Rate for PTX Series Packet Transport Routers on page 617
- Understanding CoS CLI Configuration Statements on PTX Series Routers on page 645
- How Schedulers Define Output Queue Properties on page 254
- Configuring a Scheduler on page 489
- **excess-rate on page 1107**
- **strict-priority-scheduler on page 1290**
- **traffic-control-profiles on page 1305**

Understanding CoS CLI Configuration Statements on PTX Series Routers

PTX Series Packet Transport Routers have no new Junos OS CLI configuration statements. However, some statements or statement options supported on other platforms are not supported or may not have effect on PTX Series devices. These exceptions are summarized here.

[edit chassis] Hierarchy Level

The following statement is not applicable to PTX Series Packet Transport Routers. There are always eight queues available. However, if there is a requirement to use only four of eight queues, you can do this by configuring the forwarding class to queue mapping, as appropriate.

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

The following CLI is not applicable to PICs supported on PTX Series Packet Transport Routers:

```
[edit chassis fpc slot-number pic pic-number],
   q-pic-large-buffer {
   [large-scale | small-scale]
}
```

Copyright © 2019, Juniper Networks, Inc.

On PTX Series Packet Transport Routers, buffer occupancy is computed as weighted average. However, configuration of weight at the PIC level is not supported. The default weights are applied.

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

The following CLI is not applicable to PICs supported on PTX Series Packet Transport Routers:

```
[edit chassis fpc slot-number pic pic-number],
traffic-manager {
  egress-shaping-overhead number;
  ingress-shaping-overhead number;
  mode session-shaping;
}
```

[edit class-of-service] Hierarchy Level

The following CLI is not applicable to PTX Series Packet Transport Routers because there are no separate fabric queues and egress queues:

```
fabric {
  scheduler-map {
    priority (high | low) scheduler scheduler-name;
  }
}
```

The following CLI does not support the **priority** and **policing-priority** options.

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

The following statements are not supported on PTX Series Packet Transport Routers:

- **inet-precedence** rewrite
- Rewrite of both exp and inet-precedence fields for VPN and non-VPN traffic that use the **mpls-inet-both** and **mpls-inet-both-non-vpn** protocol types.
- **exp-push-push-push** and **exp-swap-push-push** rules
- **input-scheduler-map** and **input-shaping-rate**
- The physical interface scheduler is applied on the Packet Forwarding Engine, hence the **scheduler-map-chassis** statement is not applicable.

```
interfaces {
```


Copyright © 2019, Juniper Networks, Inc.

```
  interface-name {
    input-scheduler-map map-name;
    input-shaping-rate rate;
    scheduler-map-chassis map-name;
    unit logical-unit-number {
      rewrite-rules{
        inet-precedence (rewrite-name | default) protocol
        protocol-types;
        exp (write-name | default) protocol protocol-types;
        exp-push-push-push default;
        exp-swap-push-push default;
      }
    }
  }
}
```

In the following CLI, only the **inet-precedence** statement is not supported.

```
rewrite-rules {
  (dscp | dscp-ipv6 | exp | ieee-802.1 |ieee-802.1ad |inet-precedence) rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
      loss-priority level code-point (alias | bits);
    }
  }
}
```

Classifiers on routing instances are not supported on PTX Series Packet Transport Routers because L3VPN is not supported. Hence, the following CLI is not applicable.

```
[edit class-of-service]
routing-instances routing-instance-name {
  classifiers {
    exp (classifier-name | default);
    dscp (classifier-name | default);
    dscp-ipv6 (classifier-name | default);
  }
}
```

The following limitations apply to statements under **schedulers** on PTX Series Packet Transport Routers:

- **protocol** (non-tcp | tcp) is not supported for **drop-profile-map**. The **any** option is supported.
- **excess-priority** is not supported.
- **rate-limit** is supported for **transmit-rate**. It is applied only when schedulers are configured as **strict-high**.

```
schedulers (CoS) {
  scheduler-name {
    buffer-size (percent percentage | remainder | temporal microseconds);
```

Copyright © 2019, Juniper Networks, Inc.

```
    drop-profile-map loss-priority (any | low | medium-low | medium-high
    high) protocol (any ) drop-profile profile-name;
    priority priority-level;
    transmit-rate (rate | percent percentage | remainder) <exact | rate-limit>;
  }
}
```

NOTE: Configurations that are supported only on Gigabit Ethernet IQ PICs, channelized IQ PICs, and so forth are not applicable to PTX Series Packet Transport Routers. These PICs are not supported on this platform. Those CLIs are not listed here.

[edit firewall] Hierarchy Level

In the following CLI, the **dscp** clause is not supported.

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


Copyright © 2019, Juniper Networks, Inc.

CHAPTER 16

Configuring Class of Service on T Series Core Routers

- CoS Features and Limitations on M Series and T Series Routers on page 649
- Packet Flow on Juniper Networks T Series Core Routers on page 656
- Identifying PICs Restricted to Four Queues on T Series Core Routers on page 659
- Managing Ingress Oversubscription at the PFE on page 660
- Configuring Traffic Class Maps to Manage Ingress Oversubscription on page 661
- Example: Configuring Traffic Class Maps on page 665
- Applying a Shaping Rate to Physical Interfaces Overview on page 676
- Configuring the Shaping Rate for Physical Interfaces on page 676

CoS Features and Limitations on M Series and T Series Routers

Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers, as well as M Series Multiservice Edge Routers with enhanced Flexible PIC Concentrators (FPCs), have more CoS capabilities than M Series routers that use other FPC models. Table 59 on page 554 lists some of these the differences.

To determine whether your M Series router is equipped with an enhanced FPC, issue the **show chassis hardware** command. The presence of an enhanced FPC is designated by the **E-FPC** description in the output.

```
user@host> show chassis hardware
Hardware inventory:
Item             Version  Part number Serial number     Description
Chassis                                 31959            M7i
Midplane         REV 02   710-008761   CA0209           M7i Midplane
Power Supply 0   REV 04   740-008537   PD10272          AC Power Supply
Routing Engine   REV 01   740-008846   1000396803       RE-5.0
CFEB             REV 02   750-009492   CA0166           Internet Processor IIv1
FPC 0                                                  E-FPC
  PIC 0          REV 04   750-003163   HJ6416           1x G/E, 1000 BASE-SX
  PIC 1          REV 04   750-003163   HJ6423           1x G/E, 1000 BASE-SX
  PIC 2          REV 04   750-003163   HJ6421           1x G/E, 1000 BASE-SX
  PIC 3          REV 02   750-003163   HJ0425           1x G/E, 1000 BASE-SX
FPC 1                                                  E-FPC
```

Copyright © 2019, Juniper Networks, Inc.

```
PIC 2          REV 01   750-009487   HM2275           ASP - Integrated
PIC 3          REV 01   750-009098   CA0142           2x F/E, 100 BASE-TX
```

Many operations involving the DSCP bits depend on the router and PIC type. For example, some DSCP classification configurations for MPLS and Internet can only be performed on M120 routers, M320 routers with Enhanced Type III FPCs, and MX Series routers only.

Table 59 on page 554 summarizes CoS features and limitations on M Series and T Series routers.

NOTE: The T4000 router supports the lowest of the scaling numbers for classifiers, rewrite rules, and WRED associated with MX Series and T Series routers.

Table 65: CoS Features and Limitations on M Series and T Series Routers

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
Classifiers					

Copyright © 2019, Juniper Networks, Inc.

Table 65: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
Maximum number per FPC or PIC	1	8	64	64 or 58 total	On IQ2 and IQ2E PICs, the CoS classification and CoS rewrite processes are off-loaded from the FPC to the PIC, so the capabilities and limitations of these types of PICs must be taken into consideration. For M Series router FPCs, the one-classifier limit includes the default IP precedence classifier. If you create a new classifier and apply it to an interface, the new classifier does not override the default classifier for other interfaces on the same FPC. In general, the first classifier associated with a logical interface is used. The default classifier can be replaced only when a single interface is associated with the default classifier. Only 58 user-configurable BA classifiers can be attached to logical interfaces on Type-4 FPCs in T640, T1600, or T4000 routers, because six default classifiers are automatically attached to the interfaces. When interfaces on the FPC come up, three default classifiers are installed in the FPC ASIC table: IPv4 and IPv6, MPLS tagging, and multiservices. Next, three default BA classifiers are installed: DSCP IPv6 (9), and MPLS EXP (10), and IP precedence (13). For user-defined BA classifier types (**dscp**, **dscp-ipv6**, **ieee-802.1p**, **ieee-802.1ad**, **inet-precedence**, and **mpls-exp**), you can attach a maximum of 32 classifiers of the same type (including one default classifier) to a logical interface hosted on a Type-4 FPC in a T640, T1600, or T4000 router. You can attach a maximum of 8 user-configured BA classifiers of the same type to a logical interface hosted on an Enhanced Scaling FPC in a T640, T1600, or T4000 router.
dscp	No	Yes	Yes	Yes	On all routers, you cannot configure IP precedence and DiffServ code point (DSCP) classifiers on a single logical interface, because both apply to IPv4 packets.

Copyright © 2019, Juniper Networks, Inc.

Table 65: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
dscp-ipv6	No	Yes	Yes	Yes	For T Series routers, you can apply separate classifiers for IPv4 and IPv6 packets per logical interface. For M Series router enhanced FPCs, you cannot apply separate classifiers for IPv4 and IPv6 packets. Classifier assignment works as follows: • If you assign a DSCP classifier only, IPv4 and IPv6 packets are classified using the DSCP classifier. • If you assign an IP precedence classifier only, IPv4 and IPv6 packets are classified using the IP precedence classifier. The lower three bits of the DSCP field are ignored because IP precedence mapping requires the upper three bits only. • If you assign either the DSCP or the IP precedence classifier in conjunction with the DSCP IPv6 classifier, the commit fails. • If you assign a DSCP IPv6 classifier only, IPv4 and IPv6 packets are classified using the DSCP IPv6 classifier, but the commit displays a warning message.
ieee-802.1p	No	Yes	Yes	Yes	On M Series router enhanced FPCs and T Series routers, if you associate an IEEE 802.1p classifier with a logical interface, you cannot associate any other classifier with that logical interface. For most PICs, if you apply an IEEE 802.1p classifier to a logical interface, you cannot apply non-IEEE classifiers on other logical interfaces on the same physical interface. This restriction does not apply to Gigabit Ethernet IQ2 PICs.
inet-precedence	Yes	Yes	Yes	Yes	On all routers, you cannot assign IP precedence and DSCP classifiers to a single logical interface, because both apply to IPv4 packets.
mpls-exp	Yes	Yes	Yes	Yes	For M Series router FPCs, only the default MPLS EXP classifier is supported; the default MPLS EXP classifier takes the EXP bits 1 and 2 as the output queue number.
Loss priorities based on the Frame Relay discard eligible (DE) bit	No	No	No	No	–

Copyright © 2019, Juniper Networks, Inc.

Table 65: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
Drop Profiles					
Maximum number per FPC or PIC	2	16	32	32	–
Per queue	No	Yes	Yes	Yes	–
Per loss priority	Yes	Yes	Yes	Yes	–
Per Transmission Control Protocol (TCP) bit	No	Yes	Yes	Yes	–
Policing					
Adaptive shaping for Frame Relay traffic	No	No	No	No	–
Traffic policing	Yes	Yes	Yes	Yes	–
Two-rate tricolor marking (TCM)	No	No	Yes	Yes	Allows you to configure up to four loss priorities. Two-rate TCM is supported on T Series routers with Enhanced III FPCs and the T640 Core Router with Enhanced Scaling FPC4.
Virtual channels	No	No	No	No	–
Queuing					

Copyright © 2019, Juniper Networks, Inc.

Table 65: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
Priority	No	Yes	Yes	Yes	Gigabit Ethernet IQ2 PICs support only one queue in the scheduler map with **medium-high**, **high**, or **strict-high** priority. If more than one queue is configured with **high** or **strict-high** priority, the one that appears first in the configuration is implemented as **strict-high** priority. This queue receives unlimited transmission bandwidth. The remaining queues are implemented as **low** priority, which means they might be starved. On the IQE PIC, you can rate-limit the strict-high and high queues. Without this limiting, traffic that requires low latency (delay) such as voice can block the transmission of medium-priority and low-priority packets. Unless limited, high and strict-high traffic is always sent before lower priority traffic. Support for the **medium-low** and **medium-high** queuing priority mappings varies by FPC type.
Per-queue output statistics	No	Yes	Yes	Yes	Per-queue output statistics are shown in the output of the **show interfaces queue** command.
Rewrite Markers					
Maximum number per FPC or PIC	No maximum	No maximum	64	64	On IQ2 and IQ2E PICs, the CoS classification and CoS rewrite processes are off-loaded from the FPC to the PIC, so the capabilities and limitations of these types of PICs must be taken into consideration.
dscp	No	Yes	Yes	Yes	For M Series router Enhanced FPCs, bits 0 through 5 are rewritten, and bits 6 through 7 are preserved. For M320 and T Series router non-IQ FPCs, bits 0 through 5 are rewritten, and bits 6 through 7 are preserved. For M320 and T Series router FPCs, you must decode the loss priority using the firewall filter before you can use loss priority to select the rewrite CoS value. For M320 and T Series router FPCs, Multiservices and Services PIC link services IQ interfaces (**lsq-**) do not support DSCP rewrite markers.

Copyright © 2019, Juniper Networks, Inc.

Table 65: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
dscp-ipv6	No	Yes	Yes	Yes	For M Series router Enhanced FPCs and M320 and T Series router FPCs, bits 0 through 5 are rewritten, and bits 6 through 7 are preserved. For M320 and T Series routers FPCs, you must decode the loss priority using the firewall filter before you can use loss priority to select the rewrite CoS value. For M320 and T Series router FPCs, Multiservices and Services PIC link services IQ interfaces (**lsq-**) do not support DSCP rewrite markers.
frame-relay-de	No	No	No	No	–
ieee-802.1	No	Yes	Yes	Yes	For M Series router enhanced FPCs and T Series router FPCs, fixed rewrite loss priority determines the value for bit 0; queue number (forwarding class) determines bits 1 and 2. For IQ PICs, you can only configure one IEEE 802.1 rewrite rule on a physical port. All logical ports (units) on that physical port should apply the same IEEE 802.1 rewrite rule. On T Series routers only, when you configure IEEE 802.1p rewrite marking on Gigabit Ethernet IQ, Gigabit Ethernet IQ2, Gigabit Ethernet Enhanced IQ (IQE), and Gigabit Ethernet Enhanced IQ2 (IQ2E) PICs, you cannot configure more than eight forwarding classes. This limitation does not apply to M Series routers. On M Series routers, you can configure up to 16 forwarding classes when you configure IEEE 802.1p rewrite marking on any of these PICs.
inet-precedence	Yes	Yes	Yes	Yes	For M Series router FPCs, bits 0 through 2 are rewritten, and bits 3 through 7 are preserved. For M Series router Enhanced FPCs, bits 0 through 2 are rewritten, bits 3 through 5 are cleared, and bits 6 through 7 are preserved. For M320 and T Series routers FPCs, bits 0 through 2 are rewritten and bits 3 through 7 are preserved. For M320 and T Series router FPCs, you must decode the loss priority using the firewall filter before you can use loss priority to select the rewrite CoS value.

Copyright © 2019, Juniper Networks, Inc.

Table 65: CoS Features and Limitations on M Series and T Series Routers (continued)

CoS Feature	Interface Hardware				Details
	FPCs in M120 Routers	Enhanced FPCs in M120 Routers	FPCs in M320 or T Series Routers	Type-4 or Enhanced Scaling FPCs in T Series Routers	
mpls-exp	Yes	Yes	Yes	Yes	For M320 and T Series router FPCs, you must decode the loss priority using the firewall filter before you can use loss priority to select the rewrite CoS value. For M Series routers FPCs, fixed rewrite loss priority determines the value for bit 0; queue number (forwarding class) determines bits 1 and 2.

Related Documentation

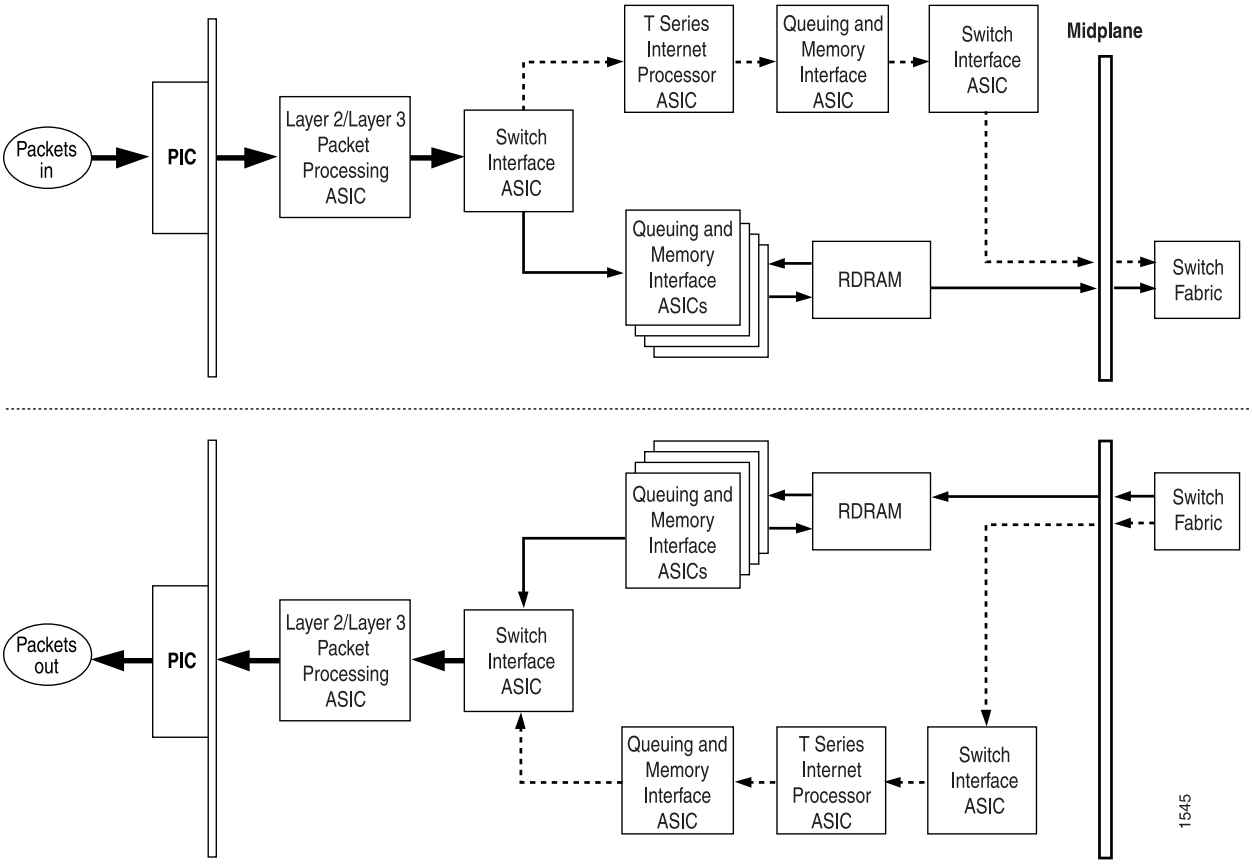
- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53
- Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 361
- Platform Support for Priority Scheduling on page 329
- CoS Features and Limitations on IQ2 and IQ2E PICs (M Series and T Series) on page 795

Packet Flow on Juniper Networks T Series Core Routers

On T Series Core Routers, CoS actions are performed in several locations: the incoming and outgoing Switch Interface ASICs, the T Series router Internet Processor ASIC, and the Queuing and Memory Interface ASICs. These locations are shown in Figure 60 on page 657.

Copyright © 2019, Juniper Networks, Inc.

Figure 60: T Series Router Packet Forwarding Engine Components and Data Flow



This topic describes the packet flow through the following components in more detail:

- Incoming Switch Interface ASICs on page 657
- T Series Routers Internet Processor ASIC on page 657
- Queuing and Memory Interface ASICs on page 658
- Outgoing Switch Interface ASICs on page 658

Incoming Switch Interface ASICs

When a data packet is passed from the receiving interface to its connected FPC, it is received by the incoming Switch Interface ASIC on that specific FPC. During the processing of the packet by this ASIC, the information in the packet's header is examined by a BA classifier. This classification action associates the packet with a particular forwarding class. In addition, the value of the packet's loss priority bit is set by this classifier. Both the forwarding class and loss priority information are placed into the notification cell, which is then transmitted to the T Series router Internet Processor ASIC.

T Series Routers Internet Processor ASIC

The T Series router Internet Processor ASIC receives notification cells representing inbound data packets and performs route lookups in the forwarding table. This lookup determines the outgoing interface on the router and the next-hop IP address for the data packet.

Copyright © 2019, Juniper Networks, Inc.

While the packet is being processed by the T Series router Internet Processor ASIC, it might also be evaluated by a firewall filter, which is configured on either the incoming or outgoing interface. This filter can perform the functions of a multifield classifier by matching on multiple elements within the packet and overwriting the forwarding class settings, loss priority settings, or both within the notification cell. Once the route lookup and filter evaluations are complete, the notification cell, now called the result cell, is passed to the Queuing and Memory Interface ASICs.

Queuing and Memory Interface ASICs

The Queuing and Memory Interface ASICs pass the data cells to memory for buffering. The data cells are placed into a queue to await transmission on the physical media. The specific queue used by the ASICs is determined by the forwarding class associated with the data packet. The configuration of the queue itself helps determine the service the packet receives while in this queued state. This functionality guarantees that certain packets are serviced and transmitted before other packets. In addition, the queue settings and the packet's loss priority setting determine which packets might be dropped from the network during periods of congestion.

In addition to queuing the packet, the outgoing I/O Manager ASIC is responsible for ensuring that CoS bits in the packet's header are correctly set before it is transmitted. This rewrite function helps the next downstream router perform its CoS function in the network.

The Queuing and Memory Interface ASIC sends the notification to the Switch Interface ASIC facing the switch fabric, unless the destination is on the same Packet Forwarding Engine. In this case, the notification is sent back to the Switch Interface ASIC facing the outgoing ports, and the packets are sent to the outgoing port without passing through the switch fabric. The default behavior is for fabric priority queuing on egress interfaces to match the scheduling priority you assign. High-priority egress traffic is automatically assigned to high-priority fabric queues.

The Queuing and Memory Interface ASIC forwards the notification, including next-hop information, to the outgoing Switch Interface ASIC.

Outgoing Switch Interface ASICs

The destination Switch Interface ASIC sends bandwidth grants through the switch fabric to the originating Switch Interface ASIC. The Queuing and Memory Interface ASIC forwards the notification, including next-hop information, to the Switch Interface ASIC. The Switch Interface ASIC sends read requests to the Queuing and Memory Interface ASIC to read the data cells out of memory, and passes the cells to the Layer 2 or Layer 3 Packet Processing ASIC. The Layer 2 or Layer 3 Packet Processing ASIC reassembles the data cells into packets, adds Layer 2 encapsulation, and sends the packets to the outgoing PIC interface. The outgoing PIC sends the packets out into the network.

Related Documentation

- Packet Flow Through the Junos OS CoS Process Overview on page 17
- Packet Flow on Juniper Networks M Series Multiservice Edge Routers on page 561
- Packet Flow on MX Series 5G Universal Routing Platforms on page 570

Copyright © 2019, Juniper Networks, Inc.

Identifying PICs Restricted to Four Queues on T Series Core Routers

Some Juniper Networks T Series Core Router PICs support up to 16 forwarding classes and are restricted to 4 queues. Contact Juniper Networks customer support for a current list of T Series router PICs that are restricted to four queues.

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

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

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

```
Qr = Qd mod Rmax
```

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

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

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

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

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

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

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

1. To map forwarding classes to restricted queues:

   ```
   [edit]
   user@host# edit class-of-service restricted-queues
   ```

2. Specify the forwarding class name and restricted queue number you want mapped.

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

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set forwarding-class class-name queue
```

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

Related Documentation

- Configuring Up to 16 Custom Forwarding Classes on page 214

Managing Ingress Oversubscription at the PFE

Ingress oversubscription is a state where the transmission rate of the incoming packets is much higher than the rate that the Packet Forwarding Engine and router can handle, causing important packets to be dropped. If an oversubscribed link or service experiences an excess of traffic, it can result in traffic loss or delay that could potentially affect other services and links.

The Packet Forwarding Engine uses fixed rules to decide the priority of incoming packets. Based on these fixed rules, the Packet Forwarding Engine categorizes incoming packets into *high-priority network control* packets and *low-priority best-effort* packets. Packets with protocols such as routing protocols are classified as *network control* packets. Packets with protocols such as Telnet, FTP, and SSH are classified as *best-effort* packets.

The limitation of these fixed rules is that even if the trusted and non-network-control packets marked by a customer edge router are forwarded to the transit router, the transit router might drop these packets. This is because, according to the fixed rules, none of these packets are high-priority packets for the transit router.

To overcome this limitation, you can prioritize and classify the traffic entering a Packet Forwarding Engine by configuring a traffic class map based on CoS values and associating the values with a traffic class such as **real-time**, **network control**, or **best-effort**. You can associate the traffic class map with an interface on the transit router. During ingress oversubscription, the router interface uses this user-defined traffic class map to select the packet priority.

NOTE: Beginning with Junos OS Release 14.2, you can configure traffic class maps on Juniper Networks T4000 Core Routers with Type 5 FPCs.

Beginning with Junos OS Release 17.2, you can configure traffic class maps on Juniper Networks MX Routers with MPCs.

Copyright © 2019, Juniper Networks, Inc.

Release History Table

Release	Description
17.2	Beginning with Junos OS Release 17.2, you can configure traffic class maps on Juniper Networks MX Routers with MPCs.
14.2	Beginning with Junos OS Release 14.2, you can configure traffic class maps on Juniper Networks T4000 Core Routers with Type 5 FPCs.

Related Documentation

- Configuring Traffic Class Maps to Manage Ingress Oversubscription on page 661

Configuring Traffic Class Maps to Manage Ingress Oversubscription

Video: Handling Ingress Oversubscription on T4000 routers with Type 5 FPCs

On T4000 routers with Type 5 FPCs and on MX Series routers with MPCs, you can prioritize and classify the traffic entering a Packet Forwarding Engine by configuring a traffic class map based on CoS code points and associating the code points with a particular traffic class, such as **network-control**, **best-effort**, or **real-time**. During ingress oversubscription, the router uses this traffic class map to identify the class type to forward or drop the packets.

To configure a traffic class map:

1. Configure the interface. This interface needs to be associated with the configured traffic class maps.

   ```
   [edit ]
   user@host# set interfaces interface-name unit unit-number family family-name address
     address
   ```

2. Create a traffic class map based on CoS code points and map the code points to a traffic class to decide the input packet priority.

   - To create a DiffServ code point (DSCP) traffic class map and map the code points to a traffic class for IPv4 and IPv6 traffic, include the following statements at the **[edit class-of-service]** hierarchy level.

     ```
     [edit class-of-service]
     user@host# set traffic-class-map dscp traffic-class-map-name traffic-class
       best-effort code-points code-point-value
     user@host# set traffic-class-map dscp traffic-class-map-name traffic-class
       network-control code-points code-point-value
     user@host# set traffic-class-map dscp traffic-class-map-name traffic-class real-time
       code-points code-point-value
     ```

   - To create an IEEE 802.1 traffic class map and map the code points to a traffic class, include the following statements at the **[edit class-of-service]** hierarchy level.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service]
user@host# set traffic-class-map ieee-802.1 traffic-class-map-name traffic-class
    best-effort code-points code-point-value
user@host# set traffic-class-map ieee-802.1 traffic-class-map-name traffic-class
    network-control code-points code-point-value
user@host# set traffic-class-map ieee-802.1 traffic-class-map-name traffic-class
    real-time code-points code-point-value
```

- To create an MPLS EXP traffic class map and map the code points to a traffic class, include the following statements at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
user@host# set traffic-class-map exp traffic-class-map-name traffic-class
    best-effort code-points code-point-value
user@host# set traffic-class-map exp traffic-class-map-name traffic-class
    network-control code-points code-point-value
user@host# set traffic-class-map exp traffic-class-map-name traffic-class real-time
    code-points code-point-value
```

- To create an IPv4 precedence traffic class map and map the code points to a traffic class, include the following statements at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
user@host# set traffic-class-map inet-precedence traffic-class-map-name
    traffic-class best-effort code-points code-point-value
user@host# set traffic-class-map inet-precedence traffic-class-map-name
    traffic-class network-control code-points code-point-value
user@host# set traffic-class-map inet-precedence traffic-class-map-name
    traffic-class real-time code-points code-point-value
```

- To create an IEEE 802.1ad code point traffic class map and map the code points to a traffic class, include the following statements at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
user@host# set traffic-class-map ieee-802.1ad traffic-class-map-name traffic-class
    best-effort code-points code-point-value
user@host# set traffic-class-map ieee-802.1ad traffic-class-map-name traffic-class
    network-control code-points code-point-value
user@host# set traffic-class-map ieee-802.1ad traffic-class-map-name traffic-class
    real-time code-points code-point-value
```

3. Associate the traffic class map with the interface that is configured in Step 1.

   - Associate the DSCP traffic class map with the interface.

```
[edit class-of-service]
user@host# set interfaces interface-name traffic-class-map dscp
    traffic-class-map-name
```

   - Associate the IEEE 802.1 traffic class map with the interface.

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


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set interfaces interface-name traffic-class-map ieee-802.1
  traffic-class-map-name <vlan-tag (inner | outer)>
```

- Associate the MPLS EXP traffic class map with the interface.

```
[edit class-of-service]
user@host# set interfaces interface-name traffic-class-map exp
  traffic-class-map-name
```

- Associate the IPv4 precedence traffic class map with the interface.

```
[edit class-of-service]
user@host# set interfaces interface-name traffic-class-map inet-precedence
  traffic-class-map-name
```

- Associate the IEEE 802.1ad traffic class map with the interface.

```
[edit class-of-service]
user@host# set interfaces interface-name traffic-class-map ieee-802.1ad
  traffic-class-map-name <vlan-tag (inner | outer)>
```

Copyright © 2019, Juniper Networks, Inc.

NOTE:

- If you do not associate the traffic class map with the configured interface, all traffic through this interface is treated with the existing fixed rule in the Packet Forwarding Engine, which prioritizes network control traffic over best-effort traffic.
- As soon as you associate a traffic class map with an interface, any code points entering that interface and not included in the traffic class map are treated as best effort.
- You can associate either an IPv4 precedence traffic class map or a DSCP traffic class map with an interface. You cannot associate both these traffic class maps with a single interface. The DSCP traffic class map applies to both IPv4 and IPv6 traffic.
- You can associate either an IEEE 802.1 traffic class map or an IEEE 802.1ad traffic class map with an interface. You cannot associate both these traffic class maps with a single interface.
- An aggregated Ethernet interface bundle can have member links from both interfaces that support traffic class maps and interfaces that do not. A configured traffic class map is associated with an aggregated Ethernet bundle in following ways:
  - If an aggregated Ethernet bundle has child links only from interfaces that support traffic class maps, then the traffic class map is associated with all links of the aggregated Ethernet bundle.
  - If an aggregated Ethernet bundle has child links only from interfaces that do not support traffic class maps, then the traffic class map is not associated with the aggregated Ethernet bundle or its links.
  - If an aggregated Ethernet bundle has child links from both interfaces that support traffic class maps and interfaces that do not, the traffic class map is associated only with the links from the interfaces that support traffic class maps.

Related Documentation

- Managing Ingress Oversubscription at the PFE on page 660
- Example: Configuring Traffic Class Maps on page 665
- traffic-class-map on page 1302
- show class-of-service forwarding-table traffic-class-map on page 1360
- show class-of-service traffic-class-map on page 1409

Copyright © 2019, Juniper Networks, Inc.

Example: Configuring Traffic Class Maps

This example shows the configuration of traffic class maps on a T4000 router with Type 5 FPC. Beginning with Junos OS Release 17.2, this example is also valid on MX Series routers with MPCs. The example is organized in the following sections:

- Requirements on page 665
- Overview and Topology on page 665
- Configuration on page 666
- Verification on page 671

Requirements

This example uses the following hardware and software components:

- One T4000 router running Junos OS Release 14.2 or later
- One customer edge (CE) router

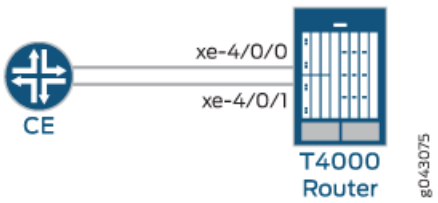
Before you configure the traffic class maps, be sure you have:

- Installed a Type 5 FPC in the T4000 router.
- Connected the CE router to the T4000 router.

Overview and Topology

This example shows the configuration of traffic class maps on a T4000 router that is connected to a CE router. The topology for this example consists of a T4000 router with Type 5 FPC connected to a CE router.

Figure 61: Configuring Traffic Class Maps on T4000 Router with Type 5 FPC



As shown in Figure 61 on page 665, the CE router forwards the traffic to interface xe-4/0/0 and xe-4/0/1 on the T4000 router.

The traffic class maps need to be configured on the T4000 router with Type 5 FPC and associated with the interface xe-4/0/0 and the interface xe-4/0/1 so that the packets can be prioritized and classified based on the user-defined configuration. When ingress oversubscription occurs, the T4000 router uses the user-defined traffic class map to process the packets.

Copyright © 2019, Juniper Networks, Inc.

This example shows how to create the following traffic class maps with CoS code points and associate these code points with the traffic class.

- IPv4 precedence traffic class map with code points 000 001, 010 011, and 100 101. Map these code points to the real-time, network-control, and best-effort traffic classes, respectively.
- MPLS EXP traffic class map with code points 000 001, 010 011, and 100 101. Map these code points to the real-time, network-control, and best-effort traffic classes, respectively.
- IEEE 802.1 traffic class map with code points 000 001, 010 011, and 100 101. Map these code points to the real-time, network-control, and best-effort traffic classes, respectively.
- DSCP traffic class map with code points 100001 100010 100011, 010011 010100 010101, and 101001 101010 101011. Map these code points to the real-time, network-control, and best-effort traffic classes, respectively.
- IEEE 802.1ad traffic class map with code points 0000 0001 1000 1001, 0010 0011 1010 1011, and 0100 0101 1100 1101. Map these code points to the real-time, network-control, and best-effort traffic classes, respectively.

The traffic class maps IPv4 precedence, MPLS EXP, and IEEE 802.1 are associated with the interface xe-4/0/0. The traffic class maps DSCP and IEEE 802.1ad are associated with the interface xe-4/0/1.

Configuration

To configure the traffic class map, perform the following tasks:

- Configuring Interfaces on page 667
- Configuring Traffic Class Maps for the Code Points and Mapping the Code Points to a Traffic Class on page 668
- Associating Interfaces with Traffic Class Maps on page 670
- Results on page 670

CLI Quick Configuration

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

```
[edit]
set interfaces xe-4/0/0 unit 0 family inet address 198.51.100.0/24
set interfaces xe-4/0/1 vlan-tagging
set interfaces xe-4/0/1 unit 0 vlan-id 111
set interfaces xe-4/0/1 unit 0 family inet address 198.51.100.1/24
set class-of-service traffic-class-map inet-precedence inetp traffic-class real-time
    code-points [ 000 001 ]
set class-of-service traffic-class-map inet-precedence inetp traffic-class network-control
    code-points [ 010 011 ]
set class-of-service traffic-class-map inet-precedence inetp traffic-class best-effort
    code-points [ 100 101 ]
```


Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service traffic-class-map exp mpls_exp traffic-class real-time code-points
  [ 000 001 ]
set class-of-service traffic-class-map exp mpls_exp traffic-class network-control
  code-points [ 010 011 ]
set class-of-service traffic-class-map exp mpls_exp traffic-class best-effort code-points
  [ 100 101 ]
set class-of-service traffic-class-map ieee-802.1 802.1p traffic-class real-time code-points
  [ 000 001 ]
set class-of-service traffic-class-map ieee-802.1 802.1p traffic-class network-control
  code-points [ 010 011 ]
set class-of-service traffic-class-map ieee-802.1 802.1p traffic-class best-effort
  code-points [ 100 101 ]
set class-of-service traffic-class-map dscp dscp_v4 traffic-class real-time code-points
  [ 100001 100010 100011 ]
set class-of-service traffic-class-map dscp dscp_v4 traffic-class network-control
  code-points [ 010011 010100 010101 ]
set class-of-service traffic-class-map dscp dscp_v4 traffic-class best-effort code-points
  [ 101001 101010 101011 ]
set class-of-service traffic-class-map ieee-802.1ad 802.1ad traffic-class real-time
  code-points [ 0000 0001 1000 1001 ]
set class-of-service traffic-class-map ieee-802.1ad 802.1ad traffic-class network-control
  code-points [ 0010 0011 1010 1011 ]
set class-of-service traffic-class-map ieee-802.1ad 802.1ad traffic-class best-effort
  code-points [ 0100 0101 1100 1101 ]
set interfaces xe-4/0/0 traffic-class-map inet-precedence inetp
set interfaces xe-4/0/0 traffic-class-map exp mpls_exp
set interfaces xe-4/0/0 traffic-class-map ieee-802.1 802.1p vlan-tag inner
set interfaces xe-4/0/1 traffic-class-map dscp dscp_v4
set interfaces xe-4/0/1 traffic-class-map ieee-802.1ad 802.1ad vlan-tag inner
```

Configuring Interfaces

Step-by-Step Procedure

Configure the interfaces. These interfaces need to be associated with traffic class maps.

- Configure the interface xe-4/0/0 with unit 0 as its logical interface, inet as protocol family, and 198.51.100.0/24 as the IP address.

```
[edit]
user@host#set interfaces xe-4/0/0 unit 0 family inet address 198.51.100.0/24
```

- Configure the interface xe-4/0/1 with unit 0 as its logical interface, inet as protocol family, and 198.51.100.1/24 as the IP address. Also, enable the VLAN tagging and configure a VLAN ID (for example, 111) to receive and transmit VLAN-tagged frames on the interface.

```
[edit]
user@host#set interfaces xe-4/0/1 vlan-tagging
user@host#set interfaces xe-4/0/1 unit 0 vlan-id 111
user@host#set interfaces xe-4/0/1 unit 0 family inet address 198.51.100.1/24
```

Copyright © 2019, Juniper Networks, Inc.

Configuring Traffic Class Maps for the Code Points and Mapping the Code Points to a Traffic Class

Step-by-Step Procedure

Copyright © 2019, Juniper Networks, Inc.

You can prioritize and classify the traffic entering a Packet Forwarding Engine by configuring a traffic class map based on the code points and associating the map with the traffic class.

- Create an IPv4 precedence traffic class map inetp and map its code points 000 001, 010 011, and 100 101 to the real-time, network control, and best-effort traffic classes, respectively.

```
[edit class-of-service]
user@host# set traffic-class-map inet-precedence inetp traffic-class real-time
   code-points [ 000 001 ]
user@host# set traffic-class-map inet-precedence inetp traffic-class network-control
   code-points [ 010 011 ]
user@host# set traffic-class-map inet-precedence inetp traffic-class best-effort
   code-points [ 100 101 ]
```

- Create an MPLS EXP traffic class map mpls_exp and map the code points 000 001, 010 011, and 100 101 to the real-time, network control, and best-effort traffic classes, respectively.

```
[edit class-of-service]
user@host# set traffic-class-map exp mpls_exp traffic-class real-time code-points
   [ 000 001 ]
user@host# set traffic-class-map exp mpls_exp traffic-class network-control
   code-points [ 010 011 ]
user@host# set traffic-class-map exp mpls_exp traffic-class best-effort code-points
   [ 100 101 ]
```

- Create a IEEE 802.1 traffic class map 802.1p and map the code points 000 001, 010 011, and 100 101 to the real-time, network control, and best-effort traffic classes, respectively.

```
[edit class-of-service]
user@host# set traffic-class-map ieee-802.1 802.1p traffic-class real-time code-points
   [ 000 001 ]
user@host# set traffic-class-map ieee-802.1 802.1p traffic-class network-control
   code-points [ 010 011 ]
user@host# set traffic-class-map ieee-802.1 802.1p traffic-class best-effort code-points
   [ 100 101 ]
```

- Create a DSCP traffic class map dscp_v4 and map the code points 100001 100010 100011, 010011 010100 010101, and 101001 101010 101011 to the real-time, network control, and best-effort traffic classes, respectively.

```
[edit class-of-service]
user@host# set traffic-class-map dscp dscp_v4 traffic-class real-time code-points [
   100001 100010 100011 ]
user@host# set traffic-class-map dscp dscp_v4 traffic-class network-control
   code-points [ 010011 010100 010101 ]
user@host# set traffic-class-map dscp dscp_v4 traffic-class best-effort code-points
   [ 101001 101010 101011 ]
```

Copyright © 2019, Juniper Networks, Inc.

- Create a IEEE802.1ad traffic class map 802.1ad and map the code points 0000 0001 1000 1001,0010 0011 1010 1011, and 0100 0101 1100 1101 to the real-time, network control, and best-effort traffic classes, respectively.

```
[edit class-of-service]
user@host# set traffic-class-map ieee-802.1ad 802.1ad traffic-class real-time
   code-points [ 0000 0001 1000 1001 ]
user@host# set traffic-class-map ieee-802.1ad 802.1ad traffic-class network-control
   code-points [ 0010 0011 1010 1011 ]
user@host# set traffic-class-map ieee-802.1ad 802.1ad traffic-class best-effort
   code-points [ 0100 0101 1100 1101 ]
```

Associating Interfaces with Traffic Class Maps

Step-by-Step Procedure

You need to associate the configured traffic class maps with the interfaces on which you want to prioritize and classify the input traffic.

- Associate the traffic class maps inetp, mpls_exp, and 802.1p with the interface xe-4/0/0.

```
[edit class-of-service]
user@host# set interfaces xe-4/0/0 traffic-class-map inet-precedence inetp
user@host# set interfaces xe-4/0/0 traffic-class-map exp mpls_exp
user@host# set interfaces xe-4/0/0 traffic-class-map ieee-802.1 802.1p vlan-tag inner
```

- Associate the traffic class map dscp_v4 and 802.1ad with the interface xe-4/0/1.

```
[edit class-of-service]
user@host# set interfaces xe-4/0/1 traffic-class-map dscp dscp_v4
user@host# set interfaces xe-4/0/1 traffic-class-map ieee-802.1ad 802.1ad vlan-tag
   inner
```

Results

```
interfaces {
   xe-4/0/0 {
      unit 0 {
         family inet {
            address 198.51.100.0/24;
         }
      }
   }
   xe-4/0/1 {
      vlan-tagging;
      unit 0 {
         vlan-id 111;
         family inet {
            address 198.51.100.1/24;
         }
      }
   }
```


Copyright © 2019, Juniper Networks, Inc.

```
}
class-of-service {
  traffic-class-map {
    inet-precedence inetp {
      traffic-class real-time code-points [ 000 001 ];
      traffic-class network-control code-points [ 010 011 ];
      traffic-class best-effort code-points [ 100 101 ];
    }
    dscp dscp_v4 {
      traffic-class real-time code-points [ 100001 100010 100011 ];
      traffic-class network-control code-points [ 010011 010100 010101 ];
      traffic-class best-effort code-points [ 101001 101010 101011 ];
    }
    exp mpls_exp {
      traffic-class real-time code-points [ 000 001 ];
      traffic-class network-control code-points [ 010 011 ];
      traffic-class best-effort code-points [ 100 101 ];
    }
    ieee-802.1 802.1p {
      traffic-class real-time code-points [ 000 001 ];
      traffic-class network-control code-points [ 010 011 ];
      traffic-class best-effort code-points [ 100 101 ];
    }
    ieee-802.1ad 802.1ad {
      traffic-class real-time code-points [ 0000 0001 1000 1001 ];
      traffic-class network-control code-points [ 0010 0011 1010 1011 ];
      traffic-class best-effort code-points [ 0100 0101 1100 1101 ];
    }
  }
  interfaces {
    xe-4/0/0 {
      traffic-class-map {
        inet-precedence inetp;
        exp mpls_exp;
        ieee-802.1 802.1p vlan-tag inner;
      }
    }
    xe-4/0/1 {
      traffic-class-map {
        dscp dscp_v4;
        ieee-802.1ad 802.1ad vlan-tag inner;
      }
    }
  }
}
```

Verification

- Verifying Mapping of Code Points to Input Traffic Classes on page 672
- Verifying Mapping of Interfaces to Traffic Class Maps on page 674
- Verifying Traffic Class Information on the Interface on page 674

Copyright © 2019, Juniper Networks, Inc.

Verifying Mapping of Code Points to Input Traffic Classes

Purpose Verify that the code points of traffic class maps are mapped to the corresponding traffic classes.

Copyright © 2019, Juniper Networks, Inc.

Action In operational mode, enter the **show class-of-service traffic-class-map** command.

```
user@host> show class-of-service traffic-class-map

Traffic-class-map: inetp, Code-point type: inet-precedence, Index: 43854
  Code point         Traffic class
  000                real-time
  001                real-time
  010                network-control
  011                network-control
  100                best-effort
  101                best-effort

Traffic-class-map: dscp_v4, Code-point type: dscp, Index: 37469
  Code point         Traffic class
  010011             network-control
  010100             network-control
  010101             network-control
  100001             real-time
  100010             real-time
  100011             real-time
  101001             best-effort
  101010             best-effort
  101011             best-effort

Traffic-class-map: mpls_exp, Code-point type: exp, Index: 39622
  Code point         Traffic class
  000                real-time
  001                real-time
  010                network-control
  011                network-control
  100                best-effort
  101                best-effort

Traffic-class-map: 802.1p, Code-point type: ieee-802.1, Index: 13605
  Code point         Traffic class
  000                real-time
  001                real-time
  010                network-control
  011                network-control
  100                best-effort
  101                best-effort

Traffic-class-map: 802.1ad, Code-point type: ieee-802.1ad, Index: 13677
  Code point         Traffic class
  0000               real-time
  0001               real-time
  0010               network-control
  0011               network-control
  0100               best-effort
  0101               best-effort
  1000               real-time
  1001               real-time
  1010               network-control
  1011               network-control
  1100               best-effort
  1101               best-effort
```

Copyright © 2019, Juniper Networks, Inc.

Meaning The display output fields **Traffic-class-map** and **Code-point type** indicate the configured traffic class map and the type of code point information, respectively.

The fields **Code point** and **Traffic class** show the mapping between the code points and the traffic class.

Verifying Mapping of Interfaces to Traffic Class Maps

Purpose Verify that the configured interfaces are mapped to the corresponding traffic class maps.

Action In operational mode, enter the **show class-of-service forwarding-table traffic-class-map mapping** command.

```
user@host> show class-of-service forwarding-table traffic-class-map mapping

Interface      Index    Table Index   Table type
xe-4/0/0         162        43854     INET-Precedence
                            39622     MPLS EXP
                            13605     IEEE-802.1
xe-4/0/1         163        37469     DSCP
                            13677     IEEE-802.1AD
```

Meaning The output shows that:

- Interface **xe-4/0/0** is associated with the traffic class maps **INET-Precedence**, **MPLS EXP**, and **IEEE-802.1**.
- Interface **xe-4/0/1** is associated with the traffic class maps **DSCP** and **IEEE-802.1AD**.

Verifying Traffic Class Information on the Interface

Purpose Verify the packet information based on the configured traffic class map.

Copyright © 2019, Juniper Networks, Inc.

Action In operational mode, enter the **show interfaces xe-4/0/0 extensive** and **show interfaces xe-4/0/1 extensive** commands.

```
user@host> show interfaces xe-4/0/0 extensive

Physical interface: xe-4/0/0, Enabled, Physical link is Up
  Interface index: 162, SNMP ifIndex: 541, Generation: 165
 Link-level type: Ethernet, MTU: 1518, MRU: 0, LAN-PHY mode, Speed: 10Gbps, BPDU
Error: None, MAC-REWRITE Error: None, Loopback: None, Source filtering: Disabled,

  Flow control: Enabled
  ...
  Preclassifier statistics:
    Traffic Class        Received Packets   Transmitted  Packets      Dropped
Packets
    real-time                     3000              3000                 0
    network-control               2000              2000                 0
    best-effort                   2000              1000                 1000

  Interface transmit statistics: Enabled

  ...
```

```
user@host> show interfaces xe-4/0/1 extensive

Physical interface: xe-4/0/1, Enabled, Physical link is Up
  Interface index: 163, SNMP ifIndex: 525, Generation: 166
 Link-level type: Ethernet, MTU: 1518, MRU: 0, LAN-PHY mode, Speed: 10Gbps, BPDU
Error: None, MAC-REWRITE Error: None, Loopback: None, Source filtering: Disabled,

  Flow control: Enabled
  ...
  Preclassifier statistics:
    Traffic Class        Received Packets   Transmitted  Packets      Dropped
Packets
    real-time                     2000              2000                 0
    network-control               1000              1000                 0
    best-effort                   1000              600                  400
  Interface transmit statistics: Enabled

    ...
```

Meaning The **Preclassifier statistics** field shows the information for received, transmitted, and dropped packets for each of the configured traffic class map.

Copyright © 2019, Juniper Networks, Inc.

Release History Table

Release	Description
17.2	Beginning with Junos OS Release 17.2, this example is also valid on MX Series routers with MPCs.

Related Documentation

- Managing Ingress Oversubscription at the PFE on page 660
- Configuring Traffic Class Maps to Manage Ingress Oversubscription on page 661
- **show class-of-service forwarding-table traffic-class-map on page 1360**
- **show class-of-service traffic-class-map on page 1409**

Applying a Shaping Rate to Physical Interfaces Overview

On T4000 routers with Type 5 FPCs and on EX Series switches, you can configure physical interfaces to shape traffic based on the rate-limited bandwidth of the total interface bandwidth. This allows you to shape the output of the physical interface, so that the interface transmits less traffic than it is physically capable of carrying.

If you do not configure a shaping rate on the physical interface, the default physical interface bandwidth is based on the channel bandwidth and the time slot allocation.

In general, the physical interface speed is the basis for calculating the various queue parameters for a physical interface such as delay buffer size, weighted round- robin (WRR) weight, drop profile, and so forth. However, when you apply a shaping rate by including the **shaping-rate** statement, the shaping rate on that physical interface becomes the basis for calculating all the queue parameters for that physical interface.

On T4000 routers with Type 5 FPCs, the shaping rate value for the physical interface must be a minimum of 292 Kbps. The maximum value of shaping rate is limited by the maximum transmission rate of the interface.

Related Documentation

- Configuring the Shaping Rate for Physical Interfaces on page 676

Configuring the Shaping Rate for Physical Interfaces

To configure the shaping rate on the physical interface, either include the **shaping-rate** statement at the **[edit class-of-service interfaces *interface-name*]** hierarchy level or include the **output-traffic-control-profile** statement at the **[edit class-of-service interfaces *interface-name*]** hierarchy level.

You can specify a peak bandwidth rate in bps, 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). For physical interfaces, the range is from 1000 through 6,400,000,000,000 bps.

Copyright © 2019, Juniper Networks, Inc.

For physical interfaces on T4000 routers with Type 5 FPCs, the shaping rate value for the physical interface must be a minimum of 292 Kbps. The maximum value of **shaping-rate** is limited by the maximum transmission rate of the interface.

The following are two example configurations for applying a shaping rate of 5 Gbps on a T4000 12x10 Gbps physical interface (xe-4/0/0):

Applying a shaping rate at the [**edit class-of-service interfaces** ***interface-name***] hierarchy:

```
[edit class-of-service]
interfaces {
   xe-4/0/0 {
      shaping-rate 5g;
   }
}
```

Applying a shaping rate using traffic-control-profiles:

```
[edit class-of-service]
traffic-control-profiles {
   output {
      shaping-rate 5g;
   }
}
interfaces {
   xe-4/0/0 {
      output-traffic-control-profile output;
   }
}
```

To view the results of your configuration, issue the following **show** commands:

- **show class-of-service interface** ***interface-name***
- **show interfaces** ***interface-name*** **extensive**

Related Documentation

- Applying a Shaping Rate to Physical Interfaces Overview on page 676

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

PART 4

Configuring Line Card-Specific and Interface-Specific Functionality

- Feature Support of Line Cards and Interfaces on page 681
- Configuring Class of Service for Tunnels on page 685
- Configuring Class of Service on Services PICs on page 697
- Configuring Class of Service on IQ and Enhanced IQ (IQE) PICs on page 721
- Configuring Class of Service on Ethernet IQ2 and Enhanced IQ2 PICs on page 793
- Configuring Class of Service on ATM Interfaces on page 841
- Configuring Class of Service on SONET/SDH OC48/STM16 IQE PICs on page 861
- Configuring Class of Service on 10-Gigabit Ethernet LAN/WAN PICs with SFP+ on page 897
- Configuring Class of Service on Enhanced Queuing DPCs on page 907
- Configuring Class of Service on MICs, MPCs, and MLCs on page 929
- Configuring Class of Service on Aggregated, Channelized, and Gigabit Ethernet Interfaces on page 1015

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 17

Feature Support of Line Cards and Interfaces

- CoS Features of the Router Hardware, PIC, MIC, and MPC Interface Families on page 681
- Scheduling on the Router Hardware, PIC, MIC, and MPC Interface Families on page 681
- Schedulers on the Router Hardware, PIC, MIC, and MPC Families on page 682
- Queuing Parameters for the Router Hardware, PIC, MIC, and MPC Interface Families on page 683

CoS Features of the Router Hardware, PIC, MIC, and MPC Interface Families

Table 66 on page 681 compares the PIC families with regard to major CoS features. Note that this table reflects the ability to perform the CoS function *at the PIC , MIC, or MPC interface level* and not on the system as a whole.

Table 66: CoS Features of the Router Hardware and Interface Families Compared

Feature:	M320 and T Series	MIC and MPC Interfaces	IQ PICs	IQ2 PICs	IQ2E PICs	Enhanced IQ PICs
BA classification	Yes	Yes	–	–	–	Yes
ToS bit rewrites	Yes	Yes	Yes, for IEEE bits only	Yes, for IEEE bits only	Yes, for IEEE bits only	–
Ingress ToS bit rewrites	–	Yes, with firewall filter	–	–	–	Yes
Hierarchical policers	–	Yes	–	–	–	Yes

Scheduling on the Router Hardware, PIC, MIC, and MPC Interface Families

Table 67 on page 682 compares the PIC, MIC, and MPC interface families with regard to scheduling abilities or features. Note that this table reflects the ability to perform the

Copyright © 2019, Juniper Networks, Inc.

function *at the PIC, MIC, or MPC interface level* and not necessarily on the system as a whole.

In this table, the OSE PICs refer to the 10-port 10-Gigabit OSE PICs (described in some guides as the 10-Gigabit Ethernet LAN/WAN PICs with SFP+).

Table 67: Scheduling on the Router Hardware and Interface Families Compared

Scheduling Feature:	M320 and T Series	MIC and MPC Interfaces	IQ PICs	IQ2 PICs	IQ2E PICs	OSE PICs on T Series	Enhanced IQ PICs
Per–unit scheduling	–	Yes, for EQ MPC	Yes	Yes	Yes	–	Yes
Physical port and logical unit shaping	–	Yes	–	Yes	Yes	–	Yes
Guaranteed rate or peak rate support	–	Yes	–	Yes, supports both CIR and PIR on the same logical unit.	Yes	Yes, at the queue level	Yes, at the logical unit
Excess rate support	–	Yes	–	–	–	Yes	Yes, at the logical unit
Shared scheduler support	–	–	–	Yes	Yes	–	–

Schedulers on the Router Hardware, PIC, MIC, and MPC Families

Table 68 on page 682 compares the PIC, MIC, and MPC interface families with regard to scheduler statements or features. Note that this table reflects the ability to perform the scheduler function *at the PIC, MIC, or MPC interface level* and not necessarily on the system as a whole.

In this table, the OSE PICs refer to the 10-port 10-Gigabit OSE PICs (described in some guides as the 10-Gigabit Ethernet LAN/WAN PICs with SFP+).

Table 68: Schedulers on the Router Hardware and Interface Families Compared

Scheduler Statement or Feature:	M320 and T Series	MIC and MPC Interfaces	IQ PICs	IQ2 PICs	IQ2E PICs	OSE PICs on T Series	Enhanced IQ PICs
Exact transmit rate	Yes	Yes	Yes	–	–	Yes	Yes
Rate-limit transmit rate	–	Yes	–	Yes	Yes	Yes	Yes

Copyright © 2019, Juniper Networks, Inc.

Table 68: Schedulers on the Router Hardware and Interface Families Compared (continued)

Scheduler Statement or Feature:	M320 and T Series	MIC and MPC Interfaces	IQ PICs	IQ2 PICs	IQ2E PICs	OSE PICs on T Series	Enhanced IQ PICs
More than one high-priority queue	Yes	Yes	Yes	–	Yes	–	Yes
Excess priority or sharing	–	Yes	–	–	–	–	Yes
Hierarchical Scheduling	–	Yes, for EQ MPC	–	–	Yes	–	–

Queuing Parameters for the Router Hardware, PIC, MIC, and MPC Interface Families

Table 69 on page 683 compares the PIC, MIC, and MPC interface families with regard to queuing parameters and features. In this table, the OSE PICs refer to the 10-port 10-Gigabit OSE PICs (described in some guides as the 10-Gigabit Ethernet LAN/WAN PICs with SFP+).

Table 69: Queue Parameters on the Router Hardware and Interface Families Compared

Queuing Statement or Feature:	M320 and T Series	MIC and MPC Interfaces	IQ PICs	IQ2 PICs	IQ2E PICs	OSE PICs on T Series	Enhanced IQ PICs
Maximum number of queues	8	8	8 on M320 or T Series routers, 4 on M7, M10, M20 routers	8	8	4 ingress, 8 egress	8
Maximum delay buffer bandwidth	80 ms: Type 1 and 2 FPC, 50 ms: Type 3 FPC	100 ms for 1 Gbps and up; 500 ms for others	100 ms	200 ms	200 ms	–	up to 4000 ms
Packet transmit priority level	3 and 3	3 and 2	2 and 2	2	3	2	3 and 2

Copyright © 2019, Juniper Networks, Inc.

Table 69: Queue Parameters on the Router Hardware and Interface Families Compared (continued)

Queuing Statement or Feature:	M320 and T Series	MIC and MPC Interfaces	IQ PICs	IQ2 PICs	IQ2E PICs	OSE PICs on T Series	Enhanced IQ PICs
Maximum number of drop profiles	32 (32 samples)	64	32 (32 samples)	32	32	–	64
Packet loss priority level	4	4	4	4	4	4	4

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 18

Configuring Class of Service for Tunnels

- CoS for Tunnels Overview on page 685
- Tunneling and BA Classifiers on page 686
- Configuring CoS for GRE and IP-IP Tunnels on page 687
- Example: Configuring CoS for GRE or IP-IP Tunnels on page 688
- Configuring a GRE Tunnel to Copy ToS Bits to the Outer IP Header on page 695

CoS for Tunnels Overview

For Multiservices and Services PIC, Link Services, and Tunnel PICs installed on Juniper Networks M Series Multiservice Edge Routers and T Series Core Routers with enhanced Flexible PIC Concentrators (FPCs), class-of-service (CoS) information is preserved inside generic routing encapsulation (GRE) and IP-IP tunnels.

For the ES PIC installed on M Series and T Series routers with enhanced FPCs, class-of-service information is preserved inside IP Security (IPsec) tunnels. For IPsec tunnels, you do not need to configure CoS, because the ES PIC copies the type-of-service (ToS) byte from the inner IP header to the GRE or IP-IP header.

For IPsec tunnels, the IP header type-of-service (ToS) bits are copied to the outer IPsec header at encryption side of the tunnel. You can rewrite the outer ToS bits in the IPsec header using a rewrite rule. On the decryption side of the IPsec tunnel, the ToS bits in the IPsec header are not written back to the original IP header field. You can still apply a firewall filter to the ToS bits to apply a packet action on egress. For more information about ToS bits and the Multiservices PICs, see "Multiservices PIC ToS Translation" on page 707. For more information about IPsec and Multiservices PICs, see the *Junos OS Services Interfaces Library for Routing Devices*.

To configure CoS for tunnels, include the following statements at the **[edit class-of-service]** and **[edit interfaces]** hierarchy level:

```
[edit class-of-service]
interfaces {
   interface-name {
      unit logical-unit-number {
         rewrite-rules {
            dscp (rewrite-name | default);
            dscp-ipv6 (rewrite-name | default);
```

Copyright © 2019, Juniper Networks, Inc.

```
            exp (rewrite-name | default)protocol protocol-types;
            exp-push-push-push default;
            exp-swap-push-push default;
            ieee-802.1 (rewrite-name | default);
            inet-precedence (rewrite-name | default);
          }
        }
      }
    }
    rewrite-rules {
      (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) rewrite-name {
        import (rewrite-name | default);
        forwarding-class class-name {
          loss-priority level code-point (alias | bits);
        }
      }
    }
    [edit interfaces]
    gre-interface-name {
      unit logical-unit-number {
        copy-tos-to-outer-ip-header;
        copy-tos-to-outer-ip-header-transit;
        tunnel {
          traffic-class traffic-class;
        }
      }
    }
```

Related Documentation

- Configuring CoS for GRE and IP-IP Tunnels on page 687
- Example: Configuring CoS for GRE or IP-IP Tunnels on page 688

Tunneling and BA Classifiers

BA classifiers can be used with GRE and IP-IP tunnels on the following routers and switches:

- EX Series switches
- M7i and M10i routers
- M Series routers with E-FPC or EP-FPC
- M120 routers
- M320 routers
- MX routers
- T Series routers

NOTE: MPCs do not support BA classifiers on gr- interfaces. Use multifield classifiers instead.

Copyright © 2019, Juniper Networks, Inc.

When a GRE or IP-IP tunnel is configured on an incoming (core-facing) interface, the queue number and PLP information are carried through the tunnel. At the egress (customer-facing) interface, the packet is queued and the CoS bits rewritten based on the information carried through the tunnel.

If no BA classifier is configured in the incoming interface, the default classifier is applied. If no rewrite rule is configured, the default rewrite rule is applied.

NOTE: For GRE and IP-IP tunnels, IP precedence and DSCP rewrite marking of the inner header do not work with more than eight forwarding classes.

Configuring CoS for GRE and IP-IP Tunnels

To configure CoS for GRE and IP-IP tunnels, perform the following configuration tasks:

1. To configure CoS for an IP-IP tunnel, include the **tunnel** statement at the **[edit interfaces ip-*fpc/pic/port* unit *logical-unit-number*]** hierarchy level. To configure CoS for a GRE tunnel, include the **tunnel** statement at the **[edit interfaces gr-*fpc/pic/port* unit *logical-unit-number*]** hierarchy level.

2. To rewrite traffic on the outbound interface, include the **rewrite-rules** statement at the **[edit class-of-service]** and **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy levels. For GRE and IP-IP tunnels, you can configure IP precedence and DSCP rewrite rules.

3. To classify traffic on the inbound interface, you can configure a behavior aggregate (BA) classifier or firewall filter. Include the **loss-priority** and **forwarding-class** statements at the **[edit firewall filter *filter-name* term *term-name* then]** hierarchy level, or the **classifiers** statement at the **[edit class-of-service]** hierarchy level.

4. For a GRE tunnel, the default is to set the ToS bits in the outer IP header to all 0s. To copy the ToS bits from the inner IP header to the outer, include the copy-tos-to-outer-ip-header-transit statement at the **[edit interfaces gr-*fpc/pic/port* unit *logical-unit-number*]** hierarchy level. (This inner-to-outer ToS bits copying is already the default behavior for IP-IP tunnels.)

   To verify that this option is enabled at the interface level, use the **show interfaces *interface-name* detail** command.

   To set a static ToS/Traffic Class value in the outer IP header, include the traffic-class ***traffic-class-value*** statement at the **[edit interfaces gr-*fpc/pic/port* unit *logical-unit-number*]** hierarchy level. Setting this value overrides the copy-tos-to-outer-ip-header-transit statement. If rewrite rules are configured on the egress WAN interface, those rewrite rules will overwrite this setting. Therefore the **traffic-class** setting only makes sense when no rewrite rules are configured.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- CoS for Tunnels Overview on page 685

Example: Configuring CoS for GRE or IP-IP Tunnels

This topic provides an example of how to configure class of service (CoS) for GRE or IP-IP tunnels.

- Requirements on page 688
- Overview on page 688
- Configuration on page 688
- Configuring Router A on page 690
- Configuring Router B on page 693
- Configuring Router C on page 694
- Verification on page 695

Requirements

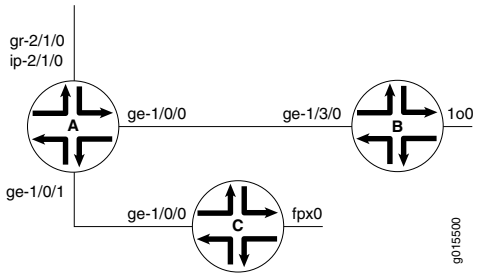
No special configuration beyond device initialization is required before configuring this example.

Overview

Topology

In Figure 62 on page 688, Router A acts as a tunnel ingress device. The link between interfaces **ge-1/0/0** in Router A and **ge-1/3/0** in Router B is the GRE or IP-IP tunnel. Router A monitors the traffic received from interface **ge-1/3/0**. By way of interface **ge-1/0/0**, Router C generates traffic to Router B.

Figure 62: CoS with a Tunnel Configuration



Configuration

To configure this example, perform these tasks:

- [xref target has no title]

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

Copyright © 2019, Juniper Networks, Inc.

configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

Router A

```
set interfaces ge-1/0/0 unit 0 family inet address 10.80.0.2/24
set interfaces ge-1/0/1 unit 0 family inet filter input zf-catch-all
set interfaces ge-1/0/1 unit 0 family inet address 10.90.0.2/24
set interfaces gr-2/1/0 unit 0 tunnel source 10.11.11.11 destination 10.255.245.46
set interfaces gr-2/1/0 unit 0 family inet address 10.21.21.21/24
set interfaces ip-2/1/0 unit 0 tunnel source 10.12.12.12 destination 10.255.245.46
set interfaces ip-2/1/0 unit 0 family inet address 10.22.22.22/24
set routing-options static route 10.1.1.1/32 next-hop gr-2/1/0.0
set routing-options static route 10.2.2.2/32 next-hop ip-2/1/0.0
set class-of-service interfaces ge-1/0/0 unit 0 rewrite-rules inet-precedence
  zf-tun-rw-ipprec-00
set class-of-service rewrite-rules inet-precedence zf-tun-rw-ipprec-00 forwarding-class
  best-effort loss-priority low code-point 000
set class-of-service rewrite-rules inet-precedence zf-tun-rw-ipprec-00 forwarding-class
  best-effort loss-priority high code-point 001
set class-of-service rewrite-rules inet-precedence zf-tun-rw-ipprec-00 forwarding-class
  expedited-forwarding loss-priority low code-point 010
set class-of-service rewrite-rules inet-precedence zf-tun-rw-ipprec-00 forwarding-class
  expedited-forwarding loss-priority high code-point 011
set class-of-service rewrite-rules inet-precedence zf-tun-rw-ipprec-00 forwarding-class
  assured-forwarding loss-priority low code-point 100
set class-of-service rewrite-rules inet-precedence zf-tun-rw-ipprec-00 forwarding-class
  assured-forwarding loss-priority high code-point 101
set class-of-service rewrite-rules inet-precedence zf-tun-rw-ipprec-00 forwarding-class
  network-control loss-priority low code-point 110
set class-of-service rewrite-rules inet-precedence zf-tun-rw-ipprec-00 forwarding-class
  network-control loss-priority high code-point 111
set class-of-service rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class best-effort
  loss-priority low code-point 000000
set class-of-service rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class best-effort
  loss-priority high code-point 001001
set class-of-service rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  expedited-forwarding loss-priority low code-point 010010
set class-of-service rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  expedited-forwarding loss-priority high code-point 011011
set class-of-service rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  assured-forwarding loss-priority low code-point 100100
set class-of-service rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  assured-forwarding loss-priority high code-point 101101
set class-of-service rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  network-control loss-priority low code-point 110110
set class-of-service rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  network-control loss-priority high code-point 111111
set firewall filter zf-catch-all term term1 then loss-priority high
set firewall filter zf-catch-all term term1 then forwarding-class network-control
```

Router B

```
user@router-B# set interfaces ge-1/3/0 unit 0 family inet address 10.80.0.1/24
user@router-B# set interfaces lo0 unit 0 family inet address 10.255.245.46/32
```

Copyright © 2019, Juniper Networks, Inc.

Router C

```
set interfaces ge-1/0/0 unit 0 family inet address 10.90.0.1/24
set routing-options static route 10.1.1.1/32 next-hop 10.90.0.2
set routing-options static route 10.2.2.2/32 next-hop 10.90.0.2
```

Configuring Router A

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure router A:

1. Configure the device interfaces.

```
[edit interfaces]
user@router-A# set ge-1/0/0 unit 0 family inet address 10.80.0.2/24
user@router-A# set ge-1/0/1 unit 0 family inet filter input zf-catch-all
user@router-A# set ge-1/0/1 unit 0 family inet address address 10.90.0.2/24
user@router-A# set gr-2/1/0 unit 0 tunnel source 10.11.11.11 destination 10.255.245.46
user@router-A# set gr-2/1/0 unit 0 family inet address 10.21.21.21/24
user@router-A# set ip-2/1/0 unit 0 tunnel source 10.12.12.12 destination 10.255.245.46
user@router-A# set ip-2/1/0 unit 0 family inet address 10.22.22.22/24
```

2. Configure the static routes.

```
[edit routing-options static]
user@router-A# set static route 10.1.1.1/32 next-hop gr-2/1/0.0
user@router-A# set static route 10.2.2.2/32 next-hop ip-2/1/0.0
```

3. Apply the rewrite rule to the interface.

```
[edit class-of-service]
user@router-A# set interfaces ge-1/0/0 unit 0 rewrite-rules inet-precedence
  zf-tun-rw-ipprec-00
```

4. Define the rewrite rules.

```
[edit class-of-service]
user@router-A# set rewrite-rules inet-precedence zf-tun-rw-ipprec-00
  forwarding-class best-effort loss-priority low code-point 000
user@router-A# set rewrite-rules inet-precedence zf-tun-rw-ipprec-00
  forwarding-class best-effort loss-priority high code-point 001
user@router-A# set rewrite-rules inet-precedence zf-tun-rw-ipprec-00
  forwarding-class expedited-forwarding loss-priority low code-point 010
user@router-A# set rewrite-rules inet-precedence zf-tun-rw-ipprec-00
  forwarding-class expedited-forwarding loss-priority high code-point 011
user@router-A# set rewrite-rules inet-precedence zf-tun-rw-ipprec-00
  forwarding-class assured-forwarding loss-priority low code-point 100
user@router-A# set rewrite-rules inet-precedence zf-tun-rw-ipprec-00
  forwarding-class assured-forwarding loss-priority high code-point 101
```


Copyright © 2019, Juniper Networks, Inc.

```
user@router-A# set rewrite-rules inet-precedence zf-tun-rw-ipprec-00
  forwarding-class network-control loss-priority low code-point 110
user@router-A# set rewrite-rules inet-precedence zf-tun-rw-ipprec-00
  forwarding-class network-control loss-priority high code-point 111
user@router-A# set rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  best-effort loss-priority low code-point 000000
user@router-A# set rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  best-effort loss-priority high code-point 001001
user@router-A# set rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  expedited-forwarding loss-priority low code-point 010010
user@router-A# set rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  expedited-forwarding loss-priority high code-point 011011
user@router-A# set rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  assured-forwarding loss-priority low code-point 100100
user@router-A# set rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  assured-forwarding loss-priority high code-point 101101
user@router-A# set rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  network-control loss-priority low code-point 110110
user@router-A# set rewrite-rules dscp zf-tun-rw-dscp-00 forwarding-class
  network-control loss-priority high code-point 111111
```

5. Configure the firewall filter.

```
[edit firewall]
user@router-A# set filter zf-catch-all term term1 then loss-priority high
user@router-A# set filter zf-catch-all term term1 then forwarding-class
  network-control
```

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

```
user@router-A# show interfaces
ge-1/0/0 {
  unit 0 {
    family inet {
      address 10.80.0.2/24;
    }
  }
}
ge-1/0/1 {
  unit 0 {
    family inet {
      filter {
        input zf-catch-all;
      }
      address 10.90.0.2/24;
    }
  }
}
```

Copyright © 2019, Juniper Networks, Inc.

```
gr-2/1/0 {
  unit 0 {
    tunnel {
      source 10.11.11.11;
      destination 10.255.245.46;
    }
    family inet {
      address 10.21.21.21/24;
    }
  }
}
ip-2/1/0 {
  unit 0 {
    tunnel {
      source 10.12.12.12;
      destination 10.255.245.46;
    }
    family inet {
      address 10.22.22.22/24;
    }
  }
}
```

```
user@router-A# show routing-options
static {
  route 10.1.1.1/32 next-hop gr-2/1/0.0;
  route 10.2.2.2/32 next-hop ip-2/1/0.0;
}
```

```
user@router-A# show class-of-service
interfaces {
  ge-1/0/0 {
    unit 0 {
      rewrite-rules {
        inet-precedence zf-tun-rw-ipprec-00;
      }
    }
  }
}
rewrite-rules {
  inet-precedence zf-tun-rw-ipprec-00 {
    forwarding-class best-effort {
      loss-priority low code-point 000;
      loss-priority high code-point 001;
    }
    forwarding-class expedited-forwarding {
      loss-priority low code-point 010;
      loss-priority high code-point 011;
    }
    forwarding-class assured-forwarding {
      loss-priority low code-point 100;
      loss-priority high code-point 101;
    }
    forwarding-class network-control {
```


Copyright © 2019, Juniper Networks, Inc.

```
        loss-priority low code-point 110;
        loss-priority high code-point 111;
      }
    }
}
dscp zf-tun-rw-dscp-00 {
    forwarding-class best-effort {
      loss-priority low code-point 000000;
      loss-priority high code-point 001001;
    }
    forwarding-class expedited-forwarding {
      loss-priority low code-point 010010;
      loss-priority high code-point 011011;
    }
    forwarding-class assured-forwarding {
      loss-priority low code-point 100100;
      loss-priority high code-point 101101;
    }
    forwarding-class network-control {
      loss-priority low code-point 110110;
      loss-priority high code-point 111111;
    }
}
```

```
user@router-A# show firewall
filter zf-catch-all {
    term term1 {
      then {
        loss-priority high;
        forwarding-class network-control;
      }
    }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Configuring Router B

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure router B:

1. Configure the device interfaces.

```
[edit interfaces]
user@router-B# set ge-1/3/0 unit 0 family inet address 10.80.0.1/24
user@router-B# set lo0 unit 0 family inet address 10.255.245.46/32
```

Copyright © 2019, Juniper Networks, Inc.

Results From configuration mode, confirm your configuration by entering the **show interfaces** command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

Router B

```
user@router-B# show interfaces
ge-1/3/0 {
   unit 0 {
      family inet {
         address 10.80.0.1/24;
      }
   }
}
lo0 {
   unit 0 {
      family inet {
         address 10.255.245.46/32;
      }
   }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Configuring Router C

Step-by-Step Procedure To configure router C:

1. Configure the device interfaces.

```
[edit interfaces]
user@router-B# set ge-1/0/0 unit 0 family inet address 10.90.0.1/24
```

2. Configure the static routes.

```
[edit routing-options static]
user@router-A# set static route 10.1.1.1/32 next-hop 10.90.0.2
user@router-A# set static route 10.2.2.2/32 next-hop 10.90.0.2
```

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

Router C

```
user@router-C show interfaces
ge-1/0/0 {
   unit 0 {
      family inet {
         address 10.90.0.1/24;
      }
   }
}
```


Copyright © 2019, Juniper Networks, Inc.

```
user@router-C show routing-options
static {
   route 10.1.1.1/32 next-hop 10.90.0.2;
   route 10.2.2.2/32 next-hop 10.90.0.2;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

To verify the configuration, run the following commands:

- **show class-of-service rewrite-rule**
- *show firewall*

Related Documentation

- CoS for Tunnels Overview on page 685
- Configuring CoS for GRE and IP-IP Tunnels on page 687
- Tunneling and BA Classifiers on page 686

Configuring a GRE Tunnel to Copy ToS Bits to the Outer IP Header

Unlike IP-IP tunnels, GRE tunnels do not copy the ToS bits to the outer IP header by default. To copy the inner ToS bits to the outer IP header (which is required for some tunneled routing protocols) on packets sent by the Routing Engine, include the **copy-tos-to-outer-ip-header** statement at the logical unit hierarchy level of a GRE interface. To copy the inner ToS bits to the outer IP header on packets transiting the device (MX Series routers with MPCs only), include the **copy-tos-to-outer-ip-header-transit** statement at the logical unit hierarchy level of a GRE interface.

To copy the inner ToS bits to the outer IP header on a GRE tunnel:

- Specify the interface on which to enable the inner IP header's ToS bits to be copied to the outer IP packet header

```
[edit]
user@host# edit interfaces
user@host# set gr-0/0/0 unit 0 copy-tos-to-outer-ip-header
user@host# set gr-0/0/0 unit 0 copy-tos-to-outer-ip-header-transit
user@host# set gr-0/0/0 unit 0 family inet
```

You can also copy the inner ToS bits to the outer IP header on packets transiting the device on a global basis for all GRE interfaces on MPCs by including the **copy-tos-to-outer service-type gre** statement at the **[edit chassis]** hierarchy level. This statement affects all GRE interfaces on MPCs and takes precedence over the **copy-tos-to-outer-ip-header-transit** statement. Once committed, this configuration

Copyright © 2019, Juniper Networks, Inc.

statement only affects new gr- interfaces. To affect an existing gr- interface, you must delete and re-add the interface.

To verify that this option is enabled at the interface level, use the **show interfaces** ***interface-name*** **detail** command.

Related Documentation

- CoS for Tunnels Overview on page 685
- Configuring CoS for GRE and IP-IP Tunnels on page 687

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 19

Configuring Class of Service on Services PICs

- CoS on Services PICs Overview on page 697
- Configuring CoS Rules on Services PICs on page 699
- Configuring CoS Rule Sets on Services PICs on page 705
- Example: Configuring CoS Rules on Services PICs on page 705
- Packet Rewriting on Services Interfaces on page 707
- Multiservices PIC ToS Translation on page 707
- Fragmentation by Forwarding Class Overview on page 708
- Configuring Fragmentation by Forwarding Class on page 709
- Configuring Drop Timeout Interval for Fragmentation by Forwarding Class on page 710
- Example: Configuring Fragmentation by Forwarding Class on page 712
- Allocating Excess Bandwidth Among Frame Relay DLCIs on Multiservices PICs on page 716
- Configuring Rate Limiting and Sharing of Excess Bandwidth on Multiservices PICs on page 718

CoS on Services PICs Overview

On Multiservices PICs with **lsq-** interfaces, there are additional features you can configure. One such feature is an additional method of classifying traffic flows based on applications, for example stateful firewalls and network address translation (NAT).

Application-based traffic flow classification enables you to configure a rule-based service that provides DiffServ code point (DSCP) marking and forwarding-class assignments for traffic transiting the Multiservices PIC. The service enables you to specify matching by application, application set, source, destination address, and match direction, and uses a similar structure to other rule-based services such as stateful firewall. The service actions allow you to associate the DSCP alias or value, forwarding-class name, system log activity, or a preconfigured application profile with the matched packet flows.

Copyright © 2019, Juniper Networks, Inc.

NOTE: If you configure a forwarding class map associating a forwarding class with a queue number, these maps are not supported on Multiservices link services intelligent queuing (lsq-) interfaces.

To configure class-of-service (CoS) features on the Multiservices PIC, include the **cos** statement at the **[edit services]** hierarchy level:

```
[edit services]
cos {
  application-profile profile-name {
    ftp {
      data {
        dscp (alias | bits);
        forwarding-class class-name;
      }
    }
    sip {
      video {
        dscp (alias | bits);
        forwarding-class class-name;
      }
      voice {
        dscp (alias | bits);
        forwarding-class class-name;
      }
    }
  }
  rule rule-name {
    match-direction (input | output | input-output);
    term term-name {
      from {
        application-sets [ set-names ];
        applications [ application-names ];
        destination-address address;
        destination-prefix-list list-name <except>;
        source-address address;
        source-address-range source-address-range low minimum-value
          high maximum-value <except>;
        source-prefix-list list-name <except>;
      }
      then {
        application-profile profile-name;
        dscp (alias | bits);
        forwarding-class class-name;
        syslog;
        reflexive; | revert; | reverse { {
          application-profile profile-name;
          dscp (alias | bits);
          forwarding-class class-name;
          syslog;
        }
      }
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
  }
  rule-set rule-set-name {
    [ rule rule-names ];
  }
}
```

Related Documentation

- Configuring CoS Rules on Services PICs on page 699
- Configuring CoS Rule Sets on Services PICs on page 705
- Example: Configuring CoS Rules on Services PICs on page 705

Configuring CoS Rules on Services PICs

This topic describes how to configure CoS rules on Services PICs.

Each CoS rule consists of a set of terms, similar to those in a firewall filter configuration. A term consists of the following:

- **from** statement—Specifies the match conditions and applications that are included and excluded.
- **then** statement—Specifies the actions and action modifiers to be performed by the router software.

If you omit the **from** term, the router accepts all traffic and the default protocol handlers take effect:

- User Datagram Protocol (UDP), Transmission Control Protocol (TCP), and Internet Control Message Protocol (ICMP) create a bidirectional flow with a predicted reverse flow.
- IP creates a unidirectional flow.

In addition, each rule must include a **match-direction** statement that specifies the direction in which the rule match is applied. To configure where the match is applied, include the **match-direction** statement at the **[edit services cos rule *rule-name*]** hierarchy level:

```
match-direction (input | output | input-output);
```

If you configure **match-direction input-output**, bidirectional rule creation is allowed.

The match direction is used with respect to the traffic flow through the Services PIC. When a packet is sent to the Services PIC, direction information is carried along with it.

On interface service sets, packet direction is determined by whether a packet is entering or leaving the interface on which the service set is applied.

With a next-hop service set, packet direction is determined by the interface used to route the packet to the Services PIC. If the inside interface is used to route the packet, the packet direction is **input**. If the outside interface is used to direct the packet to the Services

Copyright © 2019, Juniper Networks, Inc.

PIC, the packet direction is **output**. For more information on inside and outside interfaces, see *Configuring Service Sets to be Applied to Services Interfaces*.

On the Services PIC, a flow lookup is performed. If no flow is found, rule processing is performed. All rules in the service set are considered. During rule processing, the packet direction is compared against rule directions. Only rules with direction information that matches the packet direction are considered.

You can use either the source address or the destination address as a match condition, in the same way that you would configure a firewall filter; for more information, see the *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide*.

You can also include application protocol definitions that you have configured at the **[edit applications]** hierarchy level; for more information, see the *Junos OS Services Interfaces Library for Routing Devices*.

- To apply one or more specific application protocol definitions, include the **applications** statement at the **[edit services cos rule *rule-name* term *term-name* from]** hierarchy level.
- To apply one or more sets of application protocol definitions you have defined, include the **application-sets** statement at the **[edit services cos rule *rule-name* term *term-name* from]** hierarchy level.

NOTE: If you include a statement that specifies application protocols, the router derives port and protocol information from the corresponding configuration at the **[edit applications]** hierarchy level; you cannot specify these properties as match conditions.

The following sections describe how to configure CoS rules in more detail:

- Configuring Match Conditions in a CoS Rule on page 700
- Configuring Actions in a CoS Rule on page 702

Configuring Match Conditions in a CoS Rule

This topic describes how to configure the match conditions for CoS rules.

Before you begin, make sure you have completed the following tasks:

- Configure the application protocol definitions at the **[edit applications]** hierarchy level; for more information, see the **application** and *Junos OS Services Interfaces Library for Routing Devices*.
- Configure a destination prefix list by including the **prefix-list** statement at the **[edit policy-options]** hierarchy level.
- Configure a source prefix list by including the **prefix-list** statement at the **[edit policy-options]** hierarchy level.

To configure the match conditions for a CoS rule:

1. Create the CoS rule by specifying a name for it.

Copyright © 2019, Juniper Networks, Inc.

```
[edit]
user@host# edit services cos rule rule-name
```

2. Specify the direction in which the rule match is applied.

```
[edit services cos rule rule-name]
user@host# set match-direction (input | output | input-output)
```

3. Specify the input conditions for the CoS term:

   a. Define one or more target application sets.

```
[edit services cos rule rule-name]
user@host# set term term-name from application-sets [ set-names ]
```

NOTE: You must configure the application protocol definitions at the **[edit applications]** hierarchy level; for more information, see the *Junos OS Services Interfaces Library for Routing Devices*.

   b. Define one or more applications to which the CoS services apply.

```
[edit services cos rule rule-name]
user@host# set term term-name from applications [ application-names ]
```

   c. Specify the destination address for rule matching.

```
[edit services cos rule rule-name]
user@host# set term term-name from destination-address address
```

   d. Specify the name of the destination prefix list for rule matching.

```
[edit services cos rule rule-name]
user@host# set term term-name from destination-prefix-list list-name <except>
```

NOTE: You must configure the destination prefix list by including the **prefix-list** statement at the **[edit policy-options]** hierarchy level.

   e. Specify the source address for rule matching.

```
[edit services cos rule rule-name]
user@host# set term term-name from source-address address
```

   f. Specify the source address range for rule matching.

```
[edit services cos rule rule-name]
user@host# set term term-name from source-address-range source-address-range
  low minimum-value high maximum-value <except>
```

Copyright © 2019, Juniper Networks, Inc.

g. Specify the source prefix list for rule matching.

```
[edit services cos rule rule-name]
user@host# set term term-name from source-prefix-list list-name <except>
```

NOTE: You must configure the source prefix list by including the **prefix-list** statement at the **[edit policy-options]** hierarchy level.

Configuring Actions in a CoS Rule

The principal CoS actions are:

- **dscp**—Marks the packet with the specified DiffServ code point (DSCP) value or alias.
- **forwarding-class**—Assigns the packet to the specified forwarding class.

This section describes how to configure these CoS actions and includes the following topics:

- Configuring Application Profiles on page 702
- Configuring Reflexive and Reverse CoS Actions on page 703

Configuring Application Profiles

You can optionally define one or more application profiles for inclusion in CoS actions.

The **application-profile** statement includes two main components and three traffic types: **ftp** with the **data** traffic type and **sip** with the **video** and **voice** traffic types. You can set the appropriate **dscp** and **forwarding-class** values for each component within the application profile.

NOTE: The **ftp** and **sip** statements are not supported on Juniper Network MX Series 5G Universal Routing Platforms.

You can apply the application profile to a CoS configuration by including it at the **[edit services cos rule *rule-name* term *term-name* then]** hierarchy level.

To configure an application profile for inclusion in CoS actions:

1. Specify the **application-profile** statement at the **[edit services cos]** hierarchy level.

```
[edit]
user@host# edit services cos application-profile profile-name
```

2. Specify the appropriate **dscp** and **forwarding-class** value for FTP traffic.

```
[edit services cos application-profile profile-name]
user@host# set ftp data dscp (alias | bits)
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set ftp data forwarding-class class-name
```

3. Specify the appropriate **dscp** and **forwarding-class** value for SIP video traffic.

```
[edit services cos application-profile profile-name]
user@host# set sip video dscp (alias | bits)
user@host# set sip video forwarding-class class-name
```

4. Specify the appropriate **dscp** and **forwarding-class** value for SIP voice traffic.

```
[edit services cos application-profile profile-name]
user@host# set sip voice dscp (alias | bits)
user@host# set sip voice forwarding-class class-name
```

Configuring Reflexive and Reverse CoS Actions

It is important to understand that CoS services are unidirectional. It might be necessary to specify different treatments for flows in opposite directions.

Regardless of whether a packet matches the input, output, or input-output direction, flows in both directions are created. The difference is that a forward, reverse, or forward-and-reverse CoS action is associated with each flow. You should bear in mind that the flow in the opposite direction might end up having a CoS action associated with it, which you have not specifically configured.

To control the direction in which service is applied, separate from the direction in which the rule match is applied, you can configure the **reflexive** or **reverse** statement at the **[edit services cos rule *rule-name* term *term-name* then]** hierarchy level.

These two actions are mutually exclusive. If nothing is specified, data flows inherit the CoS behavior of the forward control flow.

- **reflexive** causes the equivalent reverse CoS action to be applied to flows in the opposite direction.
- **reverse** allows you to define the CoS behavior for flows in the reverse direction.

To control the direction in which a service is applied:

1. Define the CoS term actions.

```
[edit]
user@host# edit services cos rule rule-name term term-name then
```

2. Specify the action.

```
[edit services cos rule rule-name term term-name then]
user@host# set reflexive; | revert; | reverse {
```

Copyright © 2019, Juniper Networks, Inc.

3. Specify the application profile name.

```
[edit services cos rule rule-name term term-name then]
user@host# set application-profile profile-name
```

4. Define the Differentiated Services code point (DSCP) mapping that is applied to the packets.

```
[edit services cos rule rule-name term term-name then]
user@host# set dscp (alias | bits)
```

5. Define the forwarding class to which packets are assigned.

```
[edit services cos rule rule-name term term-name then]
user@host# set forwarding-class class-name
```

6. (Optional) Set the configuration to record information in the system logging facility.

   Define the forwarding class to which packets are assigned.

```
[edit services cos rule rule-name term term-name then]
user@host# set syslog
```

Related Documentation

- *Class of Service Overview*
- *Restrictions and Cautions for CoS Configuration on Services Interfaces*
- *Configuring CoS Rule Sets*
- *Examples: Configuring CoS on Services Interfaces*
- CoS on Services PICs Overview on page 697
- Configuring CoS Rule Sets on Services PICs on page 705
- Example: Configuring CoS Rules on Services PICs on page 705

Copyright © 2019, Juniper Networks, Inc.

Configuring CoS Rule Sets on Services PICs

You can define a collection of CoS rules that determine what actions the router software performs on packets in the data stream. Junos OS processes the rules in the order in which you specify them in the configuration. If a term in a rule matches the packet, the router performs the corresponding action and the rule processing stops. If no term in a rule matches the packet, processing continues to the next rule in the rule set. If none of the rules match the packet, the packet is dropped by default.The **rule-set** statement defines a collection of CoS rules that determine what actions the router software performs on packets in the data stream. You define each rule by specifying a rule name and configuring terms. You then specify the order of the rules by including the **rule-set** statement at the **[edit services cos]** hierarchy level:

This topic explains how to configure a set of CoS rules.

Before starting this procedure, make sure you define the terms and actions for the CoS rules in this rule set under the **[edit services cos rule rule-name]** hierarchy level.

To configure a CoS rule set:

1. Specify a name for the CoS rule set.

```
[edit]
user@host# edit services cos rule-set rule-set-name
```

2. Specify the name of each rule you want included in the rule set.

NOTE: Junos OS processes the rules in the order in which you specify them in the configuration.

```
[edit services cos rule-set rule-set-name]
user@host# set rule rule-name1
user@host# set rule rule-name2
```

Related Documentation

- CoS on Services PICs Overview on page 697
- Configuring CoS Rules on Services PICs on page 699
- Example: Configuring CoS Rules on Services PICs on page 705

Example: Configuring CoS Rules on Services PICs

The following example show a CoS configuration containing two rules, one for input matching on a specified application set and the other for output matching on a specified source address:

Copyright © 2019, Juniper Networks, Inc.

```
[edit services]
cos {
   application-profile cosprofile {
      ftp {
         data {
            dscp af11;
            forwarding-class 1;
         }
      }
   }
   application-profile cosrevprofile {
      ftp {
         data {
            dscp af22;
         }
      }
   }
   rule cosrule {
      match-direction input;
      term costerm {
         from {
            source-address {
               any-unicast;
            }
            applications junos-ftp;
         }
         then {
            dscp af33;
            forwarding-class 3;
            application-profile cosprofile;
            reverse {
               dscp af43;
               application-profile cosrevprofile;
            }
         }
      }
   }
}
stateful-firewall {
   rule r1 {
      match-direction input;
      term t1 {
         from {
            application-sets junos-algs-outbound;
         }
         then {
            accept;
         }
      }
      term t2 {
         then {
            accept;
         }
      }
   }
```


Copyright © 2019, Juniper Networks, Inc.

```
    service-set test {
        stateful-firewall-rules r1;
        cos-rules cosrule;
        interface-service {
            service-interface sp-1/3/0;
        }
    }
}
```

Related Documentation

- Configuring CoS Rules on Services PICs on page 699
- Configuring CoS Rule Sets on Services PICs on page 705

Packet Rewriting on Services Interfaces

On M Series routers, you can configure rewrite rules to change packet header information and attach it to an output interface. Because these rules can possibly overwrite the DSCP marking configured on Multiservices and Services PICs, it is important to create system-wide configurations carefully.

For example, knowing that the Services or Multiservices PICs can mark packets with any ToS or DSCP value and the output interface is restricted to only eight DSCP values, rewrite rules on the output interface condense the mapping from 64 to 8 values with overall loss of granularity. In this case, you have the following options:

- Remove rewrite rules in the output interface.
- Configure the output interface to include the most important mappings.

Related Documentation

- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388
- Configuring Rewrite Rules on page 391

Multiservices PIC ToS Translation

By default, all logical (**lsq-**) interfaces on a Multiservices PIC preserve the type-of-service (ToS) bits in an incoming packet header.

However, you can use the **translation-table** statement at the **[edit class-of-service]** hierarchy level to replace the arriving ToS bit pattern with a user-defined value.

This feature follows exactly the same configuration rules as the Enhanced IQ PIC. For configuration details, see "Configuring ToS Translation Tables" on page 745.

Copyright © 2019, Juniper Networks, Inc.

Fragmentation by Forwarding Class Overview

For Multiservices and Services Physical Interface Card (PIC) link services IQ (LSQ) and virtual LSQ redundancy (**rlsq-**) interfaces, you can specify fragmentation properties for specific forwarding classes. Traffic on each forwarding class can be either multilink fragmented or interleaved. By default, traffic in all forwarding classes is fragmented.

If you do not configure fragmentation properties for particular forwarding classes in multilink Point-to-Point Protocol (MLPPP) interfaces, the fragmentation threshold you set at the **[edit interfaces *interface-name* unit *logical-unit-number* fragment-threshold]** hierarchy level is used for all forwarding classes within the MLPPP interface. For multilink Frame Relay (MLFR) FRF.16 interfaces, the fragmentation threshold you set at the **[edit interfaces *interface-name* mlfr-uni-nni-bundle-options fragment-threshold]** hierarchy level is used for all forwarding classes within the MLFR FRF.16 interface. If you do not set a maximum fragment size anywhere in the configuration, packets are still fragmented if they exceed the smallest maximum transmission unit (MTU) of all the links in the bundle.

To configure fragmentation by forwarding class, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
fragmentation-maps {
  map-name {
    forwarding-class class-name {
      drop-timeout milliseconds;
      fragment-threshold bytes;
      multilink-class number;
      no-fragmentation;
    }
  }
}
interfaces {
  interface-name {
    unit logical-unit-number {
      fragmentation-map map-name;
    }
  }
}
```

Related Documentation

- Configuring Fragmentation by Forwarding Class on page 709
- Example: Configuring Fragmentation by Forwarding Class on page 712
- Configuring Drop Timeout Interval for Fragmentation by Forwarding Class on page 710
- fragmentation-map on page 1142
- fragmentation-maps on page 1143

Copyright © 2019, Juniper Networks, Inc.

Configuring Fragmentation by Forwarding Class

For Multiservices and Services PIC link services IQ (LSQ) and virtual LSQ redundancy (rlsq-) interfaces only, you can configure fragmentation properties on a particular forwarding class.

To configure fragmentation properties on a specific forwarding class:

1. Specify the name of the fragmentation map and forwarding class.

```
[edit]
user@host# edit class-of-service fragmentation-maps map-name forwarding-class
  class-name
```

2. Specify how many milliseconds to wait for fragments.

```
[edit class-of-service fragmentation-maps map-name forwarding-class class-name]
user@host# set drop-timeout milliseconds
```

NOTE: If you set this value, you must also include a multilink-class value for resequencing fragments.

3. (Optional) Specify the maximum size, in bytes, for multilink packet fragments.

```
[edit class-of-service fragmentation-maps map-name forwarding-class class-name]
user@host# set fragment-threshold bytes
```

NOTE: If you set the option, you cannot configure no-fragmentation for the forwarding class.

4. Specify the multilink class assigned to this forwarding class.

```
[edit class-of-service fragmentation-maps map-name forwarding-class class-name]
user@host# set multilink-class number
```

5. (Optional) Specify that the traffic on this particular forwarding class is interleaved, rather than fragmented.

```
[edit class-of-service fragmentation-maps map-name forwarding-class class-name]
user@host# set no-fragmentation
```

6. Apply the fragmentation map to the logical interface.

Copyright © 2019, Juniper Networks, Inc.

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

Related Documentation

- Fragmentation by Forwarding Class Overview on page 708
- Example: Configuring Fragmentation by Forwarding Class on page 712
- Configuring Drop Timeout Interval for Fragmentation by Forwarding Class on page 710
- **fragmentation-map on page 1142**
- **fragmentation-maps on page 1143**

Configuring Drop Timeout Interval for Fragmentation by Forwarding Class

For **LSQ** interfaces configured for multiclass MLPPP, you can change the drop timeout interval that the interface waits for fragment resequencing by forwarding class. This feature is mutually exclusive with the **no-fragmentation** statement configured for a forwarding class.

You can also disable the fragment resequencing function altogether by forwarding class. You do this by setting the **drop-timeout** interval to 0.

The **drop-timeout** interval can also be set at the bundle level. When the **drop-timeout** interval is set to 0 at the bundle level, *none* of the individual classes forward fragmented packets. Sequencing is ignored also, and packets are forwarded in the order in which they were received. The **drop-timeout** interval value configured at the bundle level overrides the values configured at the class level.

This example configures a logical unit on an LSQ interface with a fragmentation map setting different drop timeout values for each forwarding class:

- Best effort (BE)—The value of 0 means that no resequencing of fragments takes place for BE traffic.
- Expedited Forwarding (EF)—The value of 800 ms means that the multiclass MLPPP waits 800 ms for fragment to arrive on the link for EF traffic.
- Assured Forwarding (AF)—The absence of the timeout statements means that the default timeouts of 500 ms for links at T1 and higher speeds and 1500 ms for lower speeds are in effect for AF traffic.
- Network Control (NC)—The value of 100 ms means that the multiclass MLPPP waits 100 ms for fragment to arrive on the link for NC traffic.

To configure the drop timeout interval:

1. Define the fragmentation properties for each forwarding class.

```
[edit]
user@host# edit class-of-service fragmentation-maps Timeout_Frag_Map
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set forwarding-class BE drop-timeout 0 multilink-class 3
  fragment-threshold 128
user@host# set forwarding-class EF drop-timeout 800 multilink-class 2
user@host# set forwarding-class NC drop--timeout 100 multilink-class 0
  fragment-threshold 512
user@host# set forwarding-class AF multilink-class 1 fragment-threshold 256
```

2. Apply the fragmentation map to the logical interface.

```
[edit class-of-service]
user@host# set interfaces lsq-1/0/0 unit 1 fragmentation-map Timeout_Frag_Map
```

3. Verify the configuration.

```
[edit class-of-service fragmentation-maps Timeout_Frag_Map]
user@host# show
```

```
forwarding-class {
    BE {
        fragment-threshold 128;
        multilink-class 3;
        drop-timeout 0;  # no resequencing for this forwarding class
    }
    EF {
        multilink-class 2;
        drop-timeout 800;
    }
    NC {
        fragment-threshold 512;
        multilink-class 0;
        drop-timeout 100;
    }
    AF {
        fragment-threshold 256;  # Default timeout in effect for this class
        multilink-class 1;
    }
}
```

```
[edit class-of-service]
user@host# show
```

```
interfaces {
 lsq-1/0/0 {
        unit 1 {
            fragmentation-map Tineout_frag_Map;
        }
    }
```

4. Save the configuration.

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

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Example: Configuring Fragmentation by Forwarding Class on page 712
- Configuring Fragmentation by Forwarding Class on page 709

Example: Configuring Fragmentation by Forwarding Class

This example shows you how to configure fragmentation maps for specific forwarding classes on Multiservices PICs or Services PICs.

- Requirements on page 712
- Overview on page 712
- Configuration on page 712
- Verification on page 715

Requirements

This example uses the following hardware and software components:

- Multiservices PIC or Services PIC.

Overview

Configure two logical units on an LSQ interface. The logical units use two different fragmentation maps.

Configuration

To configure fragmentation maps for specific forwarding classes, perform these tasks:

- Define the Fragmentation Maps on page 713
- Associate the Fragmentation Maps with the an MLPPP Interface on page 714

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them in 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:

Define the Fragmentation Maps

```
set class-of-service fragmentation-maps frag-map-A forwarding-class AF
    no-fragmentation
set class-of-service fragmentation-maps frag-map-A forwarding-class EF
    no-fragmentation
set class-of-service fragmentation-maps frag-map-A forwarding-class BE
    fragment-threshold 100
set class-of-service fragmentation-maps frag-map-B forwarding-class EF
    fragment-threshold 200
set class-of-service fragmentation-maps frag-map-B forwarding-class BE
    fragment-threshold 200
set class-of-service fragmentation-maps frag-map-B forwarding-class AF
    fragment-threshold 200
```


Copyright © 2019, Juniper Networks, Inc.

Associate the Fragmentation Map with an Interface

```
set class-of-service interfaces lsq-1/0/0 unit 1 fragmentation-map frag-map-A
set class-of-service interfaces lsq-1/0/0 unit 2 fragmentation-map frag-map-B
```

Define the Fragmentation Maps

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To define the fragmentation maps:

1. Specify a name for the first fragmentation map.

   ```
   [edit]
   user@host# edit class-of-service fragmentation-maps frag-map-A
   ```

2. Define the first fragmentation map.

   a. Define the fragmentation properties for the AF forwarding class to be interleaved, rather than fragmented.

      ```
      [edit class-of-service fragmentation-maps frag-map-A]
      user@ost# set forwarding-class AF no-fragmentation
      ```

   b. Define the fragmentation properties for the EF forwarding class to be interleaved, rather than fragmented.

      ```
      [edit class-of-service fragmentation-maps frag-map-A]
      user@ost# set forwarding-class EF no-fragmentation
      ```

   c. Define the fragmentation properties for the BE forwarding class to be fragmented.

      ```
      [edit class-of-service fragmentation-maps frag-map-A]
      user@ost# set forwarding-class BE fragment-threshold 100
      ```

3. Define the second fragmentation map.

   a. Specify a name for the second fragmentation map.

      ```
      [edit class-of-service fragmentation-maps]
      user@host# edit frag-map-B
      ```

   b. Define the fragmentation properties for the EF forwarding class to be fragmented.

      ```
      [edit class-of-service fragmentation-maps frag-map-B]
      user@ost# set forwarding-class EF fragment-threshold 200
      ```

   c. Define the fragmentation properties for the BE forwarding class to be fragmented.

      ```
      [edit class-of-service fragmentation-maps frag-map-B]
      user@ost# set forwarding-class BEfragment-threshold 200
      ```

Copyright © 2019, Juniper Networks, Inc.

d. Define the fragmentation properties for the AF forwarding class to be fragmented.

```
[edit class-of-service fragmentation-maps frag-map-B]
user@ost# set forwarding-class AF fragment-threshold 200
```

Results Verify the configuration of the fragmentation maps and forwarding classes.

```
[edit class-of-service fragmentation-maps]
user@host# show
```

```
frag-map-A {
    forwarding-class {
        AF {
            no-fragmentation;
        }
        EF {
            no-fragmentation;
        }
        BE {
            fragment-threshold 100;
        }
    }
}
frag-map-B {
    forwarding-class {
        EF {
            fragment-threshold 200;
        }
        BE {
            fragment-threshold 200;
        }
        AF {
            fragment-threshold 200;
        }
    }
}
```

Associate the Fragmentation Maps with the an MLPPP Interface

Step-by-Step Procedure To associate a fragmentation map with an interface:

- Associate each fragmentation map with a logical interface.

```
[edit]
user@host# edit class-of-service interfaces lsq-1/0/0
user@host# set unit 1 fragmentation-map frag-map-A
user@host# set unit 2 fragmentation-map frag-map-B
```

Results Verify that the fragmentation maps are associated with the interfaces.

```
[edit class-of-service]
user@host# show
```


Copyright © 2019, Juniper Networks, Inc.

```
interfaces {
 lsq-1/0/0 {
        unit 1 {
            fragmentation-map frag-map-A;
        }
        unit 2 {
            fragmentation-map frag-map-B;
        }
    }
```

Verification

Verifying the Fragmentation Properties

Purpose Verify the fragmentation properties for specific forwarding classes.

Action The following output displays the fragmentation properties and forwarding class association.

```
user@host> show class-of-service fragmentation-map
```

```
Fragmentation map: frag-map-A, Index: 19801
  Forwarding class: AF
  No Fragmentation

Forwarding class: EF
  No Fragmentation

  Forwarding class: BE
  Fragmentation threshold: 100

Fragmentation map: frag-map-B, Index: 19855
  Forwarding class: EF
  Fragmentation threshold: 200

Forwarding class: BE
  Fragmentation threshold: 200

  Forwarding class: AF
  Fragmentation threshold: 200
```

Meaning The output shows the forwarding class associated with each fragmentation map, as well as the fragmentation properties associated with the forwarding class.

Related Documentation

- Fragmentation by Forwarding Class Overview on page 708
- Configuring Fragmentation by Forwarding Class on page 709

Copyright © 2019, Juniper Networks, Inc.

Allocating Excess Bandwidth Among Frame Relay DLCIs on Multiservices PICs

By default, all logical (**lsq-**) interfaces on a Multiservices PIC share bandwidth equally in the excess region (that is, bandwidth available once these interfaces have exhausted their committed information rate (CIR).

However, you can include the **excess-rate** statement to control an independent set of parameters for bandwidth sharing in the excess region of a frame relay data-link connection identifier (DLCI) on a Multiservices PIC. Include the excess-rate statement at the **[edit class-of-service traffic-control-profile *traffic-control-profile-name*]** hierarchy level.

There are several limitations to this feature:

- The excess bandwidth comes from bandwidth not used by any DLCIs (that is, bandwidth above the CIR). Therefore, only FRF.16 is supported.
- Only CIR mode is supported (you must configure a CIR on at least one DLCI).
- Only the **percent** option is supported for **lsq-** interfaces. The **priority** option is not supported for DLCIs.
- You cannot configure this feature if you also include one of the following statements in the configuration:
  - **scheduler-map**
  - **shaping-rate**
- If you oversubscribe the DLCIs, then the bandwidth can only be distributed equally.
- The **excess-priority** statement is not supported. However, for consistency, this statement will not result in a commit error.
- This feature is only supported on the Multiservices 100, Multiservices 400, and Multiservices 500 PICs.

The following procedure configures excess bandwidth sharing in the ratio of 70 to 30 percent for two frame relay DLCIs. Only FRF.16 interfaces are supported.

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

1. Enable the association of scheduler map names with logical interfaces.

```
[edit]
user@host# edit interfaces lsq-1/3/0:0
user@host# set per-unit-scheduler unit 0 dlci 100
user@host# set per-unit-scheduler unit 1dlci 200
```

2. Configure the traffic control profiles.

Copyright © 2019, Juniper Networks, Inc.

NOTE: Only the percent option is supported.

```
[edit class-of-service]
user@host# set traffic-control-profiles tc_70 excess-rate percent 70
user@host# set traffic-control-profiles tc_30 excess-rate percent 30
```

3. Apply the traffic control profiles to the logical interface.

NOTE: Only FRF.16 is supported.

```
[edit]
user@host# edit interfaces lsq-1/3/0
user@host# set unit 0 output-traffic-control-profile tc_70
user@host# set unit 1 output-traffic-control-profile tc_30
```

4. Verify the configuration.

[edit interfaces lsq-1/3/0:0]
user@host# show

```
per-unit-scheduler;
unit 0 {
    dlci 100;
}
unit 1 {
    dlci 200;
}
```

[edit class-of-service]
user@host# show

```
traffic-control-profiles {
 tc_70 {
        excess-rate percent 70;
    }
    tc_30 {
        excess-rate percent 30;
    }
}
```

```
[edit interfaces]
user@host# show
lsq-1/3/0 {
  unit 0 {
    output-traffic-control-profile tc_70;
  }
  unit 1 {
    output-traffic-control-profile tc_30;
  }
}
```

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Allocating Excess Bandwidth Among Frame Relay DLCIs on Multiservices PICs on page 716

Configuring Rate Limiting and Sharing of Excess Bandwidth on Multiservices PICs

On Multiservices PICs, you can limit the transmit rate of a logical interface (**lsq-**) in the same way as other types of queuing PICs. You can also assign a percentage of the excess bandwidth to the logical interfaces. As with other types of PICs, the strict-high queue (voice) can "starve" low and medium priority queues. To prevent the strict-high queue from starving other queues, rate-limit the queue.

To rate-limit logical interfaces on a Multiservices PIC, include the **transmit-rate** statement with the **rate-limit** option at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level:

```
[edit class-of-service schedulers scheduler-name]
transmit-rate (rate | percent percentage | remainder) rate-limit;
```

You can also make the excess strict-high bandwidth available for other queues. You can split the excess bandwidth among multiple queues, but the total excess bandwidth assigned to these queues can only add up to 100 percent. The excess-bandwidth **priority** statement option is not supported on the Multiservices PIC. For more information about excess bandwidth sharing, see "Configuring Excess Bandwidth Sharing on IQE PICs" on page 753.

To share excess bandwidth among Multiservices PICs, include the **excess-rate** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level.

```
[edit class-of-service schedulers scheduler-name]
excess-rate percent percentage;
```

Both of these rate-limiting and excess bandwidth sharing features apply to egress traffic only, and only for per-unit schedulers. Hierarchical schedulers and shared schedulers are not supported.

You must still complete the configuration by configuring the scheduler map and applying it to the Multiservices PIC interface.

This example configures a rate limit and excess bandwidth sharing for a Multiservices PIC interface.

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

1. Specify the scheduler name and parameter values.

```
[edit]
user@host# edit class-of-service
user@host# set schedulers scheduler0 transmit-rate percent 10 rate-limit
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set schedulers scheduler0 priority strict-high excess-rate percent 30
user@host# set schedulers scheduler1 transmit-rate percent 1 rate-limit
user@host# set schedulers scheduler1 priority high excess-rate percent 70
```

2. Specify a scheduler map name and associate it with the scheduler configuration and forwarding class.

```
[edit class-of-service]
user@host# set scheduler-maps scheduler0 forwarding-class ef scheduler scheduler0
user@host# set scheduler-maps scheduler0 forwarding-class af scheduler scheduler1
```

3. Associate the scheduler map name with the interface.

```
[edit class-of-service]
user@host# set interfaces lsq-1/3/0 unit 0 scheduler-map scheduler0
user@host# set interfaces lsq-1/3/0 unit 1 scheduler-map scheduler1
```

4. Verify the configuration.

```
scheduler0 {
    transmit-rate {
        percent 10;
        rate-limit;
    }
    excess-rate percent 30;
    priority strict-high;
}
scheduler1 {
    transmit-rate {
        percent 1;
        rate-limit;
    }
    excess-rate percent 70;
    priority high;
}
```

```
[edit class-of-service]
user@host# show schedulers
```

```
interfaces {
 lsq-1/3/0 {
        unit 0 {
            scheduler-map scheduler0;
        }
    }
```

```
scheduler-maps {
 scheduler0 {
        forwarding-class ef scheduler scheduler0;
```

Copyright © 2019, Juniper Networks, Inc.

```
        forwarding-class af scheduler scheduler1;
    }
}
```

Related Documentation

- CoS on Services PICs Overview on page 697
- Configuring Schedulers on page 258
- Configuring Scheduler Maps on page 258
- Excess Rate and Excess Priority Configuration Examples on page 290

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 20

Configuring Class of Service on IQ and Enhanced IQ (IQE) PICs

- CoS on Enhanced IQ PICs Overview on page 721
- Calculation of Expected Traffic on IQE PIC Queues on page 722
- Configuring the Junos OS to Support Eight Queues on IQ Interfaces for T Series and M320 Routers on page 744
- BA Classifiers and ToS Translation Tables on page 745
- Configuring ToS Translation Tables on page 745
- Configuring Hierarchical Layer 2 Policers on IQE PICs on page 751
- Configuring Excess Bandwidth Sharing on IQE PICs on page 753
- Configuring Rate-Limiting Policers for High Priority Low-Latency Queues on IQE PICs on page 759
- Applying Scheduler Maps and Shaping Rate to Physical Interfaces on IQ PICs on page 761
- Applying Scheduler Maps to Chassis-Level Queues on page 778
- Assigning Default Frame Relay Rewrite Rule to IQE PICs on page 790
- Defining Custom Frame Relay Rewrite Rule on IQE PICs on page 791

CoS on Enhanced IQ PICs Overview

The Enhanced IQ (IQE) PIC family supports a series of non-channelized and channelized interfaces that run at a large variety of speeds. Sophisticated Class-of-Service (CoS) techniques are available for the IQE PICs at the channel level. These techniques include policing based on type-of-service (ToS) bits, five priority levels, two shaping rates (the guaranteed rate and shaping rate), a shared scheduling option, DiffServ code point (DSCP) rewrite on egress, and configurable delay buffers for queuing. All of these features, with numerous examples, are discussed in this chapter. For a comparison of the capabilities of IQE PICs with other types of PICs, see "CoS Features and Limitations on M Series and T Series Routers" on page 553.

For information about CoS components that apply generally to all interfaces, see "Understanding How Class of Service Manages Congestion and Controls Service Levels in the Network" on page 4. For general information about configuring interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.

Copyright © 2019, Juniper Networks, Inc.

IQE PICs can be used in Juniper Networks M40e, M120, M320 Multiservice Edge Routers and T Series Core Routers to supply enhanced CoS capabilities for edge aggregation. The same interface configuration syntax is used for basic configuration, and other CoS statements are applied at channel levels. Some configuration statements are available only in Junos OS Release 9.3 and later, as noted in this chapter.

Calculation of Expected Traffic on IQE PIC Queues

This topic discusses the following topics related to calculating the expected traffic flow on IQE PIC queues:

- Excess Bandwidth Calculations Terminology on page 722
- Excess Bandwidth Basics on page 722
- Logical Interface Modes on IQE PICs on page 724
- Default Rates for Queues on IQE PICs on page 728
- Sample Calculations of Excess Bandwidth Sharing on IQE PICs on page 730

Excess Bandwidth Calculations Terminology

The following terms are used in this discussion of IQE PIC queue calculations:

- CIR mode—A physical interface is in CIR mode when one of more of its “children” (logical interfaces in this case) have a guaranteed rate configured, but some logical interfaces have a shaping rate configured.
- Default mode—A physical interface is in default mode if none of its “children” (logical interfaces in this case) have a guaranteed rate or shaping rate configured.
- Excess mode—A physical interface is in excess mode when one of more of its “children” (logical interfaces in this case) have an excess rate configured.
- PIR mode—A physical interface is in PIR mode if none of its “children” (logical interfaces in this case) have a guaranteed rate configured, but some logical interfaces have a shaping rate configured.

Excess Bandwidth Basics

This basic example illustrates the interaction of the guaranteed rate, the shaping rate, and the excess rate applied to four queues. The same concepts extend to logical interfaces (units) and cases in which the user does not configure an explicit value for these parameters (in that case, the system uses implicit parameters).

In this section, the term “not applicable” (NA) means that the feature is not explicitly configured. All traffic rates are in megabits per second (Mbps).

The hardware parameters derived from the configured rates are relatively straightforward except for the excess weight. The excess rate is translated into an absolute value called the excess weight. The scheduler for an interface picks a logical unit first, and then a queue within the logical unit for transmission. Logical interfaces and queues that are within their guaranteed rates are picked first, followed by those in the excess region. If the transmission rate for a logical interface or queue is more than the shaping rate, the

Copyright © 2019, Juniper Networks, Inc.

scheduler skips the logical interface or queue. Scheduling in the guaranteed region uses straight round-robin, whereas scheduling in the excess region uses weighed round-robin (WRR) based on the excess weights. The excess weights are in the range from 1 to 127, but they are transparent to the user and subject to change with implementation. The weights used in this example are for illustration only.

This example uses a logical interface with a transmit rate (CIR) of 10 Mbps and a shaping rate (PIR) of 10 Mbps. The user has also configured percentage values of transmit rate (CIR), shaping rate (PIR), and excess rate as shown in Table 70 on page 723.

Table 70: Basic Example of Excess Bandwidth

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	5%	5%	10%	10 Mbps
Q1	30%	80%	50%	10 Mbps
Q2	10%	15%	30%	10 Mbps
Q3	15%	35%	30%	10 Mbps

The values used by the hardware based on these parameters are shown in Table 71 on page 723.

Table 71: Hardware Use of Basic Example Parameters

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Weight	Expected Traffic Rate
Q0	0.5 Mbps	0.5 Mbps	10	0.5 Mbps
Q1	3 Mbps	8 Mbps	50	5.19 Mbps
Q2	1 Mbps	1.5 Mbps	30	1.5 Mbps
Q3	1.5 Mbps	3.5 Mbps	30	2.81 Mbps
Totals:	6 Mbps	13.5 Mbps	120	10 Mbps (maximum output)

There are a number of important points regarding excess bandwidth calculations:

- The guaranteed rates should add up to less than the logical interface guaranteed rate (10 Mbps).
- Shaping rates (PIRs) can be oversubscribed.
- Excess rates can be oversubscribed. This rate is only a ratio at which the sharing occurs.
- Each queue receives the minimum of the guaranteed bandwidth because each queue is transmitting at its full burst if it can.

Copyright © 2019, Juniper Networks, Inc.

- The excess (remaining) bandwidth is shared among the queues in the ratio of their excess rates. In this case, the excess bandwidth is the logical interface bandwidth minus the sum of the queue transmit rates, or 10 Mbps – 6 Mbps = 4 Mbps.
- However, transmission rates are capped at the shaping rate (PIR) of the queue. For example, Queue 0 gets 0.5 Mbps.
- Queue 0 also gets a guaranteed transmit rate (CIR) of 0.5 Mbps and is eligible for excess bandwidth calculated as 4 Mbps (10 Mbps – 6 Mbps) multiplied by 10/127. However, because the shaping rate (PIR) for Queue 0 is 0.5 Mbps, the expected traffic rate is capped at 0.5 Mbps.
- Queue 1 gets its guaranteed transmit rate (CIR) of 3 Mbps. Because Queue 0 has already been dealt with, Queue 1 is eligible for sharing the excess bandwidth along with Queue 2 and Queue 3. So Queue 1 is entitled to an excess bandwidth of 4 Mbps multiplied by 50 / (30 + 30 + 50), or 1.81 Mbps.
- In the same way, Queue 2 is eligible for its guaranteed transmit rate (CIR) of 1 Mbps and an excess bandwidth of 4 Mbps multiplied by 30 / (30 + 30 + 50), or 1.09 Mbps. However, because Queue 2 has a shaping rate (PIR) of 1.5 Mbps, the bandwidth of Queue 2 is capped at 1.5 Mbps. The additional 0.59 Mbps can be shared by Queue 1 and Queue 3.
- Queue 3 is eligible for an excess of 4 Mbps multiplied by 30 / (30 + 30 + 50), or 1.09 Mbps. This total of 2.59 Mbps is still below the shaping rate (PIR) for Queue 3 (3.5 Mbps).
- The remaining bandwidth of 0.59 Mbps (which Queue 2 could not use) is shared between Queue 1 and Queue 3 in the ratio 50/30. So Queue 3 can get 0.59 multiplied by 30 / (50 + 30), or 0.22 Mbps. This gives a total of 2.81 Mbps.
- Therefore, Queue 1 gets 3 Mbps + 1.82 Mbps + (0.59 Mbps * 50 / (50 + 30)), or approximately 5.19 Mbps.

Logical Interface Modes on IQE PICs

On IQE PICs, scheduling occurs level-by-level. That is, based on the parameters configured on the logical interface, the scheduler first picks a logical interface to transmit from. Then, based on the configuration of the underlying queues, the IQE PIC selects one of the queues to transmit from. Therefore, it is important to understand how different logical interface parameters are configured or derived (not explicitly configured), and also how the same values are established at the queue level.

In the following examples, assume that the bandwidth available at the physical interface level is 400 Mbps and there are four logical interfaces (units) configured. A per-unit scheduler is configured, so the logical interfaces operate in different modes depending on the parameters configured.

If no class-of-service parameters are configured on any of the logical interfaces, the interface is in default mode. In default mode, the guaranteed rate (CIR) available at the physical interface (400 Mbps) is divided equally among the four logical interfaces. Each of the four gets a guaranteed rate (CIR) of 100 Mbps. Because none of the four logical interfaces have a shaping rate (PIR) configured, each logical interface can transmit up

Copyright © 2019, Juniper Networks, Inc.

to the maximum of the entire 400 Mbps. Because there is no excess rate configured on any of the logical interfaces, each of the four gets an equal, minimum excess weight of 1. The configured and hardware-derived bandwidths for this default mode example are shown in Table 72 on page 725.

Table 72: Default Mode Example for IQE PICs

Logical Interface	Configured			Hardware		
	Guaranteed rate (CIR)	Shaping Rate (PIR)	Excess Rate	Guaranteed Rate	Shaping Rate	Excess Weight
Unit 0	NA	NA	NA	100 Mbps	400 Mbps	1
Unit 1	NA	NA	NA	100 Mbps	400 Mbps	1
Unit 2	NA	NA	NA	100 Mbps	400 Mbps	1
Unit 3	NA	NA	NA	100 Mbps	400 Mbps	1

If a subset of the logical interfaces (units) have a shaping rate (PIR) configured, but none of them have a guaranteed rate (CIR) or excess rate, then the physical interface is in PIR mode. Furthermore, if the sum of the shaping rates on the logical interfaces is less than or equal to the physical interface bandwidth, the physical interface is in undersubscribed PIR mode. If the sum of the shaping rates on the logical interfaces is more than the physical interface bandwidth, the physical interface is in oversubscribed PIR mode. These modes are the same as on other PICs, where only a shaping rate and guaranteed rate can be configured.

In undersubscribed PIR mode, the logical interfaces with a configured shaping rate receive preferential treatment over those without a configured shaping rate. For logical interfaces with a shaping rate configured, the guaranteed rate is set to the shaping rate. For the logical interfaces without a shaping rate, the remaining logical interface bandwidth is distributed equally among them. Excess weights for the logical interfaces with a shaping rate are set to an implementation-dependent value proportional to the shaping rate. Excess weights for the logical interfaces without a shaping rate are set to the minimum weight (1). However, although the excess weights for the configured logical interfaces are never used because the logical interfaces cannot transmit above their guaranteed rates, the excess weights are still determined for consistency with oversubscribed mode. Also, logical interfaces without a configured shaping rate can transmit up to a maximum of the physical bandwidth of the other queues that are not transmitting. Therefore, the shaping rate (PIR) is set to the physical interface bandwidth on these interfaces.

The configured and hardware-derived bandwidths for the undersubscribed PIR mode example are shown in Table 73 on page 726. Note that the sum of the shaping rates configured on the logical interfaces (500 Mbps) is more than the physical interface bandwidth (400 Mbps).

Copyright © 2019, Juniper Networks, Inc.

Table 73: Undersubscribed PIR Mode Example for IQE PICs

Logical Interface	Configured			Hardware		
	Guaranteed rate (CIR)	Shaping Rate (PIR)	Excess Rate	Guaranteed Rate	Shaping Rate	Excess Weight
Unit 0	NA	100 Mbps	NA	100 Mbps	100 Mbps	127
Unit 1	NA	200 Mbps	NA	200 Mbps	200 Mbps	63
Unit 2	NA	NA	NA	50 Mbps	400 Mbps	1
Unit 3	NA	NA	NA	50 Mbps	400 Mbps	1

In the oversubscribed PIR mode, where the sum of the configured shaping rates on the logical interfaces exceeds the physical interface bandwidth, we cannot set the guaranteed rate to the shaping rate because this might result in the sum of the guaranteed rates exceeding the physical interface bandwidth, which is not possible. In this mode, we want the logical interfaces with shaping rates configured to share the traffic proportionally when these logical interfaces are transmitting at full capacity. This could not happen if the guaranteed rate was set to the shaping rate. Instead, in hardware, we set the guaranteed rates to a "scaled down" shaping rate, so that the sum of the guaranteed rates of the logical interfaces do not exceed the physical interface bandwidth. Because there is no remaining bandwidth once this is done, the other logical interfaces receive a guaranteed rate of 0. Excess weights are set proportionally to the shaping rates and for logical interfaces without a shaping rate, the excess weight is set to a minimum value (1). Finally, the shaping rate is set to the shaping rate configured on the logical interface or to the physical interface bandwidth otherwise.

NOTE: When the sum of shaping rate at a logical interface is greater than the interface's bandwidth and a rate limit is applied to one of the logical interface queues, the bandwidth limit for the queue is based on a scaled down logical interface shaping rate value rather than the configured logical interface shaping rate.

The configured and hardware-derived bandwidths for the oversubscribed PIR mode example are shown in Table 74 on page 726. Note that the sum of the shaping rates configured on the logical interfaces (300 Mbps) is less than the physical interface bandwidth (400 Mbps).

Table 74: Oversubscribed PIR Mode Example for IQE PICs

Logical Interface	Configured			Hardware		
	Guaranteed rate (CIR)	Shaping Rate (PIR)	Excess Rate	Guaranteed Rate	Shaping Rate	Excess Weight
Unit 0	NA	100 Mbps	NA	80 Mbps	100 Mbps	50

Copyright © 2019, Juniper Networks, Inc.

Table 74: Oversubscribed PIR Mode Example for IQE PICs (continued)

Logical Interface	Configured			Hardware		
	Guaranteed rate (CIR)	Shaping Rate (PIR)	Excess Rate	Guaranteed Rate	Shaping Rate	Excess Weight
Unit 1	NA	150 Mbps	NA	120 Mbps	150 Mbps	76
Unit 2	NA	250 Mbps	NA	200 Mbps	250 Mbps	127
Unit 3	NA	NA	NA	0 Mbps	400 Mbps	1

If none of the logical interfaces have an excess rate configured, but at least one of the logical interfaces has a guaranteed rate (CIR) configured, then the physical interface is in CIR mode. In this case, the guaranteed rates are set in hardware to the configured guaranteed rate on the logical interface. For logical interfaces that do not have a guaranteed rate configured, the guaranteed rate is set to 0. The hardware shaping rate is set to the value configured on the logical interface or to the full physical interface bandwidth otherwise. The excess weight is calculated proportional to the configured guaranteed rates. Logical interfaces without a configured guaranteed rate receive a minimum excess weight of 1.

The configured and hardware-derived bandwidths for the CIR mode example are shown in Table 75 on page 727. In CIR mode, the shaping rates are ignored in the excess weight calculations. So although logical unit 1 has an explicitly configured PIR and logical unit 3 does not, they both receive the minimum excess weight of 1.

Table 75: CIR Mode Example for IQE PICs

Logical Interface	Configured			Hardware		
	Guaranteed rate (CIR)	Shaping Rate (PIR)	Excess Rate	Guaranteed Rate	Shaping Rate	Excess Weight
Unit 0	50 Mbps	100 Mbps	NA	50 Mbps	100 Mbps	127
Unit 1	NA	150 Mbps	NA	0 Mbps	150 Mbps	1
Unit 2	100 Mbps	NA	NA	100 Mbps	400 Mbps	63
Unit 3	NA	NA	NA	0 Mbps	400 Mbps	1

If one of the logical interfaces has an excess rate configured, then the physical interface is in excess rate mode. Strictly speaking, this mode only matters for the calculation of excess weights on the logical interface. The hardware guaranteed and shaping rates are determined as described previously. In excess rate mode, the excess weights are set to a value based on the configured excess rate. Logical interfaces which do not have excess rates configured receive a minimum excess weight of 1.

Copyright © 2019, Juniper Networks, Inc.

NOTE: Because the excess rate only makes sense above the guaranteed rate, you cannot configure an excess rate in PIR mode (PIR mode has only shaping rates configured). You must configure at least one guaranteed rate (CIR) on a logical interface to configure an excess rate.

The excess rate is configured as a percentage in the range from 1 through 100. The configured value is used to determine the excess weight in the range from 1 through 127.

The configured and hardware-derived bandwidths for the excess rate mode example are shown in Table 76 on page 728. When an excess rate is configured on one or more logical interfaces, the shaping rate and the guaranteed rate are both ignored in the excess weight calculations. So logical unit 2 gets a minimum excess weight of 1, even though it has a guaranteed rate configured.

Table 76: Excess Rate Mode Example for IQE PICs

Logical Interface	Configured			Hardware		
	Guaranteed rate (CIR)	Shaping Rate (PIR)	Excess Rate	Guaranteed Rate	Shaping Rate	Excess Weight
Unit 0	50 Mbps	100 Mbps	20%	50 Mbps	100 Mbps	50
Unit 1	NA	150 Mbps	50%	0 Mbps	150 Mbps	127
Unit 2	100 Mbps	NA	NA	100 Mbps	400 Mbps	1
Unit 3	NA	NA	50%	0 Mbps	400 Mbps	127

Default Rates for Queues on IQE PICs

The IQE PIC operates at the queue level as well as at the logical unit level. This section discusses how the IQE PIC derives hardware values from the user configuration parameters. First, the default behavior without explicit configuration is investigated, along with the rules used to derive hardware parameters from the scheduler map configuration of the transmit rate, shaping rate, and excess rate. For more information about configuring schedulers and scheduler maps, see "How Schedulers Define Output Queue Properties" on page 254.

When you do not configure any CoS parameters, a default scheduler map is used to establish four queues: best-effort, expedited-forwarding, assured-forwarding, and network-control. Each queue has the default transmit rate, shaping rate, and excess rate shown in Table 77 on page 728.

Table 77: Default Queue Rates on the IQE PIC

Queue	Transmit Rate	Shaping Rate	Excess Rate
best-effort (Q0)	95%	100%	95%

Copyright © 2019, Juniper Networks, Inc.

Table 77: Default Queue Rates on the IQE PIC (continued)

Queue	Transmit Rate	Shaping Rate	Excess Rate
expedited-forwarding (Q1)	0%	100%	0%
assured-forwarding (Q2)	0%	100%	0%
network-control (Q3)	5%	100%	5%

When you configure a scheduler map to change the defaults, the IQE PIC hardware derives the values for each of the three major parameters: transmit rate, shaping rate, and excess rate.

The transmit rate is determined as follows:

- If a transmit rate is configured, then:
  - If the transmit rate is configured as an absolute bandwidth value, the configured value is used by the hardware.
  - If the transmit rate is configured as a percentage, then the percentage is used to calculate an absolute value used by the hardware, based on the guaranteed rate (CIR) configured at the logical interface or physical interface level. The CIR itself can be a default, configured, or derived value.
  - If the transmit rate is configured as a remainder, then the remaining value of the logical interface (unit) guaranteed rate (CIR) is divided equally among the queues configured as remainder.
- If a transmit rate is not configured, then the default transmit rate is derived based on remainder (for backward compatibility).
- If an excess rate is configured on any of the queues in a scheduler map, then the transmit rate on the queue is set to 0.

The shaping rate is determined as follows:

- If a shaping rate is configured:
  - If the shaping rate is configured as an absolute bandwidth value, the configured value is used by the hardware.
  - If the shaping rate is configured as a percentage, then the percentage is used to calculate an absolute value used by the hardware, based on the guaranteed rate (CIR) configured at the logical interface or physical interface level. Although it seems odd to base a shaping rate (PIR) on the CIR instead of a PIR, this is done so the shaping rate can be derived on the same basis as the transmit rate.
- If a shaping rate is not configured, then the default shaping rate is set to the shaping rate configured at the logical interface or physical interface level.

Copyright © 2019, Juniper Networks, Inc.

The excess rate is determined as follows:

- If an excess rate is configured on a queue, the value is used to derive an excess weight used by the IQE PIC hardware. The excess weight determines the proportional share of the excess bandwidth for which each queue can contend. The excess rate can be:
  - Percentage in the range from 1 through 100. This value is scaled to a hardware excess weight. Excess rates can add up to more than 100% for all queues under a logical or physical interface.
- If an excess rate is not configured on a queue, then the default excess rate is one of the following:
  - If a transmit rate is configured on any of the queues, then the excess weight is proportional to the transmit rates. Queues that do not have a transmit rate configured receive a minimum weight of 1.
  - If a transmit rate is not configured on any of the queues, but some queues have a shaping rate, then the excess weight is proportional to the shaping rates. Queues that do not have a shaping rate configured receive a minimum weight of 1.
  - If no parameters are configured on a queue, then the queue receives a minimum weight of 1.

Sample Calculations of Excess Bandwidth Sharing on IQE PICs

The following four examples show calculations for the PIR mode. In PIR mode, the transmit rate and shaping rate calculations are based on the shaping rate of the logical interface. All calculations assume that one logical interface (unit) is configured with a shaping rate (PIR) of 10 Mbps and a scheduler map with four queues.

The first example has only a shaping rate (PIR) configured on the queues, as shown in Table 78 on page 730.

Table 78: PIR Mode with No Excess Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	NA	80%	NA	10 Mbps
Q1	NA	50%	NA	1 Mbps
Q2	NA	40%	NA	0 Mbps
Q3	NA	30%	NA	5 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 79 on page 731.

Copyright © 2019, Juniper Networks, Inc.

Table 79: PIR Mode with No Excess Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	2.5 Mbps	8.0 Mbps	50	6 Mbps
Q1	2.5 Mbps	5.0 Mbps	31	1 Mbps
Q2	2.5 Mbps	4.0 Mbps	25	0 Mbps
Q3	2.5 Mbps	3.0 Mbps	19	3 Mbps

In this first example, all four queues are initially serviced round-robin. Because there are no transmit rates configured on any of the queues, they receive a default "remainder" transmit rate of 2.5 Mbps per queue. But because there are shaping rates configured, the excess weights are calculated based on the shaping rates. For the traffic sent to each queue, Queue 0 and Queue 3 get their transmit rates of 2.5 Mbps and Queue 1 gets 1 Mbps. The remaining 4 Mbps is excess bandwidth and is divided between Queue 0 and Queue 3 in the ratio of the shaping rates (80/30). So Queue 3 expects an excess bandwidth of 4 Mbps * (30% / (80% + 30%)) = 1.09 Mbps. However, because the shaping rate on Queue 3 is 3 Mbps, Queue 3 can transmit only 3 Mbps and Queue 0 receives the remaining excess bandwidth and can transmit at 6 Mbps.

Note that if there were equal transmit rates explicitly configured, such as 2.5 Mbps for each queue, the excess bandwidth would be split based on the transmit rate (equal in this case), as long as the result in below the shaping rate for the queue.

The second example has a shaping rate (PIR) and transmit rate (CIR) configured on the queues, as shown in Table 80 on page 731.

Table 80: PIR Mode with Transmit Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	50%	80%	NA	10 Mbps
Q1	40%	50%	NA	5 Mbps
Q2	10%	20%	NA	5 Mbps
Q3	NA	5%	NA	1 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 81 on page 731.

Table 81: PIR Mode with Transmit Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	5.0 Mbps	8.0 Mbps	63	5 Mbps

Copyright © 2019, Juniper Networks, Inc.

Table 81: PIR Mode with Transmit Rate Hardware Behavior (continued)

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q1	4.0 Mbps	5.0 Mbps	50	4 Mbps
Q2	1.0 Mbps	2.0 Mbps	12	1 Mbps
Q3	0.0 Mbps	0.5 Mbps	1	0.0 Mbps

In this second example, because the transmit rates are less than the shaping rates, each queue receives its transmit rate.

The third example also has a shaping rate (PIR) and transmit rate (CIR) configured on the queues, as shown in Table 82 on page 732.

Table 82: Second PIR Mode with Transmit Rate Configuration Example

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	50%	80%	NA	10 Mbps
Q1	40%	50%	NA	5 Mbps
Q2	5%	20%	NA	0 Mbps
Q3	NA	5%	NA	1 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 83 on page 732.

Table 83: Second PIR Mode with Transmit Rate Hardware Behavior Example

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	5.0 Mbps	8.0 Mbps	66	5.27 Mbps
Q1	4.0 Mbps	5.0 Mbps	53	4.23 Mbps
Q2	0.5 Mbps	2.0 Mbps	13	0.0 Mbps
Q3	0.5 Mbps	0.5 Mbps	1	0.5 Mbps

In this third example, all four queues are initially serviced round-robin. However, Queue 2 has no traffic sent to its queue. So Queue 0, Queue 1, and Queue 3 all get their respective transmit rates, a total of 9.5 Mbps. The remaining 0.5 Mbps is used by Queue 3, because the transmit rate is the same as the shaping rate. Once this traffic is sent, Queue 0 and Queue 1 share the excess bandwidth in the ratio of their transmit rates, which total 9 Mbps. In this case, Queue 0 = 5 Mbps + (0.5 Mbps * 5/9) = 5.27 Mbps. Queue 1 = 4 Mbps + (0.5 Mbps * 4/9) = 4.23 Mbps.

Copyright © 2019, Juniper Networks, Inc.

The fourth example has a shaping rate (PIR), transmit rate (CIR), and excess rate configured on the queues, as shown in Table 84 on page 733.

Table 84: PIR Mode with Transmit Rate and Excess Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	30%	80%	50%	10 Mbps
Q1	25%	50%	10%	5 Mbps
Q2	10%	20%	30%	0 Mbps
Q3	5%	5%	NA	1 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 85 on page 733.

Table 85: PIR Mode with Transmit Rate and Excess Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	3.0 Mbps	8.0 Mbps	70	6.33 Mbps
Q1	2.5 Mbps	5.0 Mbps	14	3.17 Mbps
Q2	1.0 Mbps	2.0 Mbps	42	0.0 Mbps
Q3	0.5 Mbps	0.5 Mbps	1	0.5 Mbps

In this fourth example, all four queues are initially serviced round-robin. Queue 3 gets 0.5 Mbps of guaranteed bandwidth but cannot transmit more because the shaping rate is the same. Queue 2 has no traffic to worry about at all. Queue 0 and Queue 1 get the respective transmit rates of 3.0 Mbps and 2.5 Mbps. The excess bandwidth of 4 Mbps is divided between Queue 0 and Queue 1 in the ratio on their excess rates. So Queue 1 gets 2.5 Mbps (the guaranteed rate) + 4 Mbps (the excess) + (10% / (50% + 10%)) = 3.17 Mbps. Queue 0 gets the rest, for a total of 6.33 Mbps.

You can configure only an excess rate on the queues, as shown in Table 86 on page 733.

Table 86: Excess Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	NA	NA	50%	10 Mbps
Q1	NA	NA	40%	10 Mbps
Q2	NA	NA	30%	10 Mbps
Q3	NA	NA	20%	10 Mbps

Copyright © 2019, Juniper Networks, Inc.

The way that the IQE PIC hardware interprets these excess rate parameters is shown in Table 87 on page 734.

Table 87: Excess Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	0 Mbps	10.0 Mbps	45	3.57 Mbps
Q1	0 Mbps	10.0 Mbps	40	2.86 Mbps
Q2	0 Mbps	10.0 Mbps	30	2.14 Mbps
Q3	0 Mbps	10.0 Mbps	20	1.43 Mbps

In this excess rate example, there are no transmit or shaping rates configured on any of the queues, only excess rates, so bandwidth division happens only on the basis of the excess rates. Note that all the transmit (guaranteed) rates are set to 0. Usually, when there are no excess rates configured, the queue transmit rate is calculated by default. But when there is an excess rate configured on any of the queues, the transmit rate is set to 0. The excess bandwidth (all bandwidths in this case) is shared in the ratio of the excess weights. So Queue 0 receives 10 Mbps * (50 / (50 + 40+ 30+ 20)) = 3.57 Mbps.

It is possible to configure rate limits that result in error conditions. For example, consider the configuration shown in Table 88 on page 734.

Table 88: PIR Mode Generating Error Condition

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	NA	80%	NA	10 Mbps
Q1	NA	50%	NA	5 Mbps
Q2	NA	20%	NA	5 Mbps
Q3	NA	5%	NA	1 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 89 on page 734.

Table 89: PIR Mode Generating Error Condition Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	2.5 Mbps	8.0 Mbps	818	4.03 Mbps
Q1	2.5 Mbps	5.0 Mbps	511	3.47 Mbps
Q2	2.5 Mbps	2.0 Mbps	255	2 Mbps

Copyright © 2019, Juniper Networks, Inc.

Table 89: PIR Mode Generating Error Condition Behavior (continued)

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q3	2.5 Mbps	0.5 Mbps	51	0.1 Mbps

In the error example, note that the shaping rates calculated on Queue 2 and Queue 3 are less than the transmit rates on those queues (2.0 Mbps and 0.5 Mbps are each less than 2.5 Mbps). This is an error condition and results in a syslog error message.

The following set of five examples involve the IQE PIC operating in CIR mode. In CIR mode, the transmit rate and shaping rate calculations are based on the transmit rate of the logical interface. All calculations assume that the logical interface has a shaping rate (PIR) of 20 Mbps and a transmit rate (CIR) of 10 Mbps. The scheduler map has four queues.

The first example has only a shaping rate (PIR) with no excess rate configured on the queues, as shown in Table 90 on page 735.

Table 90: CIR Mode with No Excess Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	NA	80%	NA	10 Mbps
Q1	NA	70%	NA	10 Mbps
Q2	NA	40%	NA	10 Mbps
Q3	NA	30%	NA	10 Mbps

NOTE: The transmit rate (CIR) of 10 Mbps is configured on the logical interface (unit) not the queues in the scheduler map. This is why the queue transmit rates are labeled NA.

The way that the IQE PIC hardware interprets these parameters is shown in Table 91 on page 735.

Table 91: CIR Mode with No Excess Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	2.5 Mbps	8.0 Mbps	50	6.76 Mbps
Q1	2.5 Mbps	7.0 Mbps	31	6.23 Mbps
Q2	2.5 Mbps	4.0 Mbps	25	4.0 Mbps
Q3	2.5 Mbps	3.0 Mbps	19	3.0 Mbps

Copyright © 2019, Juniper Networks, Inc.

In this first example, all four queues split the 10-Mbps transmit rate equally and each get a transmit rate of 2.5 Mbps. However, the shaping rate on the interface is 20 Mbps. The 10-Mbps excess bandwidth is divided among the queues in the ratio of their shaping rates. But Queue 2 and Queue 3 are shaped at 3.0 and 4.0 Mbps, respectively, so they cannot use more bandwidth and get those rates. This accounts for 2 Mbps (the 7 Mbps shaped bandwidth minus the 5 Mbps guaranteed bandwidth for Queue 2 and Queue 3) of the 10-Mbps excess, leaving 8 Mbps for Queue 0 and Queue 1. So Queue 0 and Queue 1 share the 8-Mbps excess bandwidth in the ratio of their shaping rates, which total 15 Mbps. In this case, Queue 0 = 8.0 Mbps * 8/15 = 4.26 Mbps, for a total of 2.5 Mbps + 4.26 Mbps = 6.76 Mbps. Queue 1 = 8.0 Mbps * 7/15 = 3.73 Mbps, for a total of 2.5 Mbps + 3.73 Mbps = 6.23 Mbps.

The second example has only a few shaping rates (PIR) with no excess rate configured on the queues, as shown in Table 92 on page 736.

Table 92: CIR Mode with Some Shaping Rates and No Excess Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	NA	80%	NA	10 Mbps
Q1	NA	50%	NA	5 Mbps
Q2	NA	NA	NA	10 Mbps
Q3	NA	NA	NA	1 Mbps

NOTE: If a configuration results in the calculated transmit rate of the queue exceeding the shaping rate of the queue, an error message is generated. For example, setting the shaping rate on Queue 2 and Queue 3 in the above example to 20 percent and 5 percent, respectively, generates an error message because the calculated transmit rate for these queues (2.5 Mbps) is more than their calculated shaping rates (2.0 Mbps and 0.5 Mbps).

The way that the IQE PIC hardware interprets these parameters is shown in Table 93 on page 736.

Table 93: CIR Mode with Some Shaping Rates and No Excess Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	2.5 Mbps	8.0 Mbps	78	8.0 Mbps
Q1	2.5 Mbps	5.0 Mbps	48	5.0 Mbps
Q2	2.5 Mbps	20 Mbps	1	6.0 Mbps
Q3	2.5 Mbps	20 Mbps	1	1.0 Mbps

Copyright © 2019, Juniper Networks, Inc.

In this second example, all four queues split the 10-Mbps transmit rate equally and each get a transmit rate of 2.5 Mbps. Because of their configured queue shaping rates, Queue 0 and Queue 1 receive preference over Queue 2 and Queue 3 for the excess bandwidth. Queue 0 (8.0 Mbps) and Queue 1 (5.0 Mbps) account for 13 Mbps of the 20 Mbps shaping rate on the logical interface. The remaining 7 Mbps is divided equally between Queue 2 and Queue 3. However, because Queue 3 only has 1 Mbps to send, Queue 2 uses the remaining 6 Mbps.

The third example has shaping rates (PIR) and transmit rates with no excess rate configured on the queues, as shown in Table 94 on page 737.

Table 94: CIR Mode with Shaping Rates and Transmit Rates and No Excess Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	50%	80%	NA	10 Mbps
Q1	40%	50%	NA	5 Mbps
Q2	10%	20%	NA	5 Mbps
Q3	NA	10%	NA	1 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 95 on page 737.

Table 95: CIR Mode with Shaping Rates and Transmit Rates and No Excess Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	5.0 Mbps	8.0 Mbps	63	8.0 Mbps
Q1	4.0 Mbps	5.0 Mbps	50	5.0 Mbps
Q2	1.0 Mbps	2.0 Mbps	12	2.0 Mbps
Q3	0.0 Mbps	0.5 Mbps	1	0.5 Mbps

In this third example, the first three queues get their configured transmit rates and are serviced in round-robin fashion. This adds up to 10 Mbps, leaving a 10-Mpbs excess from the logical interface shaping rate of 20 Mbps. The excess is shared in the ratio of the transmit rates, or 5:4:1:0. Therefore, Queue 0 receives 5 Mbps + (5 * 10/10) = 10 Mbps. This value is greater than the 8 Mbps shaping rate on Queue 0, so Queue 0 is limited to 8 Mbps. Queue 1 receives 4 Mbps + (4 * 10/10) = 8 Mbps. This value is greater than the 5 Mbps shaping rate on Queue 1, so Queue 1 is limited to 5 Mbps. Queue 2 receives 1 Mbps + (1 * 10/10) = 2 Mbps. This value is equal to the 2 Mbps shaping rate on Queue 2, so Queue 2 receives 2 Mbps. This still leaves 5 Mbps excess bandwidth, which can be used by Queue 3. Note that in this example bandwidth usage never reaches the shaping rate configured on the logical interface (20 Mbps).

Copyright © 2019, Juniper Networks, Inc.

The fourth example has shaping rates (PIR) and transmit rates with no excess rate configured on the queues. However, in this case the sum of the shaping rate percentages configured on the queues multiplied by the transmit rate configured on the logical interface is greater than the shaping rate configured on the logical interface. The configuration is shown in Table 96 on page 738.

Table 96: CIR Mode with Shaping Rates Greater Than Logical Interface Shaping Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	50%	80%	NA	10 Mbps
Q1	40%	70%	NA	10 Mbps
Q2	10%	50%	NA	10 Mbps
Q3	NA	50%	NA	10 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 97 on page 738.

Table 97: CIR Mode with Shaping Rates Greater Than Logical Interface Shaping Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	5.0 Mbps	8.0 Mbps	63	8.0 Mbps
Q1	4.0 Mbps	7.0 Mbps	50	7.0 Mbps
Q2	1.0 Mbps	5.0 Mbps	12	5.0 Mbps
Q3	0.0 Mbps	5.0 Mbps	1	0.0 Mbps

In this fourth example, the first three queues get their configured transmit rates and are serviced in round-robin fashion. This adds up to 10 Mbps, leaving a 10-Mpbs excess from the logical interface shaping rate of 20 Mbps. The excess is shared in the ratio of the transmit rates, or 5:4:1:0. Therefore, Queue 0 receives 5 Mbps + (5 * 10/10) = 10 Mbps. This value is greater than the 8 Mbps shaping rate on Queue 0, so Queue 0 is limited to 8 Mbps. Queue 1 receives 4 Mbps + (4 * 10/10) = 8 Mbps. This value is greater than the 7 Mbps shaping rate on Queue 1, so Queue 1 is limited to 7 Mbps. Queue 2 receives 1 Mbps + (1 * 10/10) = 2 Mbps. This value is less than the 5 Mbps shaping rate on Queue 2, so Queue 2 receives 2 Mbps. This still leaves 3 Mbps excess bandwidth, which can be used by Queue 2 (below its shaping rate) and Queue 3 (also below its shaping rate) in the ratio 1:0 (because of the transmit rate configuration). But 1:0 means Queue 3 cannot use this bandwidth, and Queue 2 utilizes 2 Mbps + (3 Mbps * 1/1) = 5 Mbps. This is equal to the shaping rate of 5 Mbps, so Queue 2 receives 5 Mbps.

The fifth example has excess rates and transmit rates, but no shaping rates (PIR) configured on the queues. The configuration is shown in Table 98 on page 739.

Copyright © 2019, Juniper Networks, Inc.

Table 98: CIR Mode with Excess Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	30%	NA	50%	10 Mbps
Q1	25%	NA	10%	10 Mbps
Q2	NA	NA	30%	10 Mbps
Q3	10%	NA	NA	10 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 99 on page 739.

Table 99: CIR Mode with Excess Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	3.0 Mbps	20 Mbps	70	10.5 Mbps
Q1	2.5 Mbps	20 Mbps	14	4.0 Mbps
Q2	0.0 Mbps	20 Mbps	42	4.5 Mbps
Q3	1.0 Mbps	20 Mbps	1	1.0 Mbps

In this fifth example, Queue 2 does not have a transmit rate configured. If there were no excess rates configured, then Queue 2 would get a transmit rate equal to the remainder of the bandwidth (3.5 Mbps in this case). However, because there is an excess rate configured on some of the queues on this logical interface, the transmit rate for Queue 2 is set to 0 Mbps. The others queues get their transmit rates and there leaves 13.5 Mbps of excess bandwidth. This bandwidth is divided among Queue 0, Queue 1, and Queue 3 in the ratio of their excess rates. So Queue 0, for example, gets 3.0 Mbps + 13.5 Mbps * (50 / (50 + 10 + 30)) = 10.5 Mbps.

Four other examples calculating expected traffic distribution are of interest. The first case has three variations, so there are six more examples in all.

- Oversubscribed PIR mode at the logical interface with transmit rates, shaping rates, and excess rates configured at the queues (this example has three variations).
- CIR mode at the logical interface (a non-intuitive case is used).
- Excess priority configured.
- Default excess priority used.

The first three examples all concern oversubscribed PIR mode at the logical interface with transmit rates, shaping rates, and excess rates configured at the queues. They all use a configuration with a physical interface having a shaping rate of 40 Mbps. The physical interface has two logical units configured, logical unit 1 and logical unit 2, with

Copyright © 2019, Juniper Networks, Inc.

a shaping rate of 30 Mbps and 20 Mbps, respectively. Because the sum of the logical interface shaping rates is more than the shaping rate on the physical interface, the physical interface is in oversubscribed PIR mode. The CIRs (transmit rates) are set to the scaled values of 24 Mbps and 16 Mbps, respectively.

Assume that logical unit 1 has 40 Mbps of traffic to be sent. The traffic is capped at 30 Mbps because of the shaping rate of 30 Mbps. Because the CIR is scaled down to 24 Mbps, the remaining 6 Mbps (30 Mbps – 24 Mbps) qualifies as excess bandwidth.

The following three examples consider different parameters configured in a scheduler map and the expected traffic distributions that result.

The first example uses oversubscribed PIR mode with only transmit rates configured on the queues. The configuration is shown in Table 100 on page 740.

Table 100: Oversubscribed PIR Mode with Transmit Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	40%	NA	NA	15 Mbps
Q1	30%	NA	NA	10 Mbps
Q2	25%	NA	NA	10 Mbps
Q3	5%	NA	NA	5 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 101 on page 740.

Table 101: Oversubscribed PIR Mode with Transmit Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	9.6 Mbps	30 Mbps	50	12 Mbps
Q1	7.2 Mbps	30 Mbps	38	9 Mbps
Q2	6.0 Mbps	30 Mbps	31	7.5 Mbps
Q3	1.2 Mbps	30 Mbps	6	1.5 Mbps

The first example has hardware queue transmit rates based on the parent (logical interface unit 1) transmit rate (CIR) value of 24 Mbps. Because there are no excess rates configured, the excess weights are determined by the transmit rates. Therefore, both the logical interface CIR and excess bandwidth are divided in the ratio of the transmit rates. This is essentially the same as the undersubscribed PIR mode and the traffic distribution should be the same. The only difference is that the result is achieved as a combination of guaranteed rate (CIR) and excess rate sharing.

Copyright © 2019, Juniper Networks, Inc.

The second example also uses oversubscribed PIR mode, but this time with only excess rate configured on the queues. In other words, the same ratios are established with excess rate percentages instead of transmit rate percentages. In this case, when excess rates are configured, queues without a specific transmit rate are set to 0 Mbps. So the entire bandwidth qualifies as excess at the queue level and the bandwidth distribution is based on the configured excess rates. The expected output rate results are exactly the same as in the first example, except the calculation is based on different parameters.

The third example also uses oversubscribed PIR mode, but with both transmit rates and excess rates configured on the queues. The configuration is shown in Table 102 on page 741.

Table 102: Oversubscribed PIR Mode with Transmit Rate and Excess Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	40%	NA	50%	15 Mbps
Q1	30%	NA	50%	12 Mbps
Q2	25%	NA	NA	8 Mbps
Q3	5%	NA	NA	5 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 103 on page 741.

Table 103: Oversubscribed PIR Mode with Transmit Rate and Excess Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	9.6 Mbps	30 Mbps	63	12.6 Mbps
Q1	7.2 Mbps	30 Mbps	63	10.2 Mbps
Q2	6.0 Mbps	30 Mbps	1	6.0 Mbps
Q3	1.2 Mbps	30 Mbps	1	1.2 Mbps

The third example has the configured queue transmit rate (CIR) divided according to the ratio of the transmit rates based on the logical interface unit 1 CIR of 25 Mbps. The rest of the excess bandwidth divided according the ratio of the excess rates. The excess 6-Mbps bandwidth is divided equally between Queue 0 and Queue 1 because the excess rates are both configured at 50%. This type of configuration is not recommended, however, because the CIR on the logical interface is a system-derived value based on the PIRs of the other logical units and the traffic distribution at the queue level is based on this value and, therefore, not under direct user control. We recommend that you either configure excess rates without transmit rates at the queue level when in PIR mode, or also define a CIR at the logical interface if you want to configure a combination of transmit rates and excess rates at the queue level. That is, you should use configurations of the CIR mode with excess rates types.

Copyright © 2019, Juniper Networks, Inc.

The fourth example uses CIR mode at the logical interface. For this example, assume that a physical interface is configured with a 40-Mbps shaping rate and logical interfaces unit 1 and unit 2. Logical interface unit 1 has a PIR of 30 Mbps and logical interface unit 2 has a PIR of 20 Mbps and a CIR of 10 Mbps. The configuration at the queue level of logical interface unit 1 is shown in Table 104 on page 742.

Table 104: CIR Mode with Transmit Rate and Excess Rate Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	40%	NA	50%	15 Mbps
Q1	30%	NA	50%	12 Mbps
Q2	25%	NA	NA	8 Mbps
Q3	5%	NA	NA	5 Mbps

The way that the IQE PIC hardware interprets these parameters is shown in Table 105 on page 742.

Table 105: CIR Mode with Transmit Rate and Excess Rate Hardware Behavior

Queue	Transmit Rate	Shaping Rate	Excess Weight	Expected Output Rate
Q0	0 Mbps	30 Mbps	63	15 Mbps
Q1	0 Mbps	30 Mbps	63	12 Mbps
Q2	0 Mbps	30 Mbps	1	1.5 Mbps
Q3	0 Mbps	30 Mbps	1	1.5 Mbps

The fourth example might be expected to divide the 40 Mbps of traffic between the two logical units in the ratio of the configured transmit rates. But note that because the logical interfaces are in CIR mode, and logical interface unit 1 does not have a CIR configured, the hardware CIR is set to 0 Mbps at the queue level. Bandwidth distribution happens based only on the excess weights. So Queue 0 and Queue 1 get to transmit up to 15 Mbps and 12 Mbps, respectively, while the remaining 3 Mbps is divided equally by Queue 2 and Queue 3.

NOTE: We recommend configuring a CIR value explicitly for the logical interface if you are configuring transmit rates and excess rates for the queues.

The fifth example associates an excess priority with the queues. Priorities are associated with every queue and propagated to the parent node (logical or physical interface). That is, when the scheduler picks a logical interface, the scheduler considers the logical interface priority as the priority of the highest priority queue under that logical interface. On the IQE PIC, you can configure an excess priority for every queue. The excess priority

Copyright © 2019, Juniper Networks, Inc.

can differ from the priority used for guaranteed traffic and applies only to traffic in the excess region. The IQE PIC has three "regular" priorities and two excess priorities (high and low, which is the default). The excess priorities are lower than the regular priorities. For more information about configuring excess bandwidth sharing and priorities, see "Configuring Excess Bandwidth Sharing on IQE PICs" on page 753.

Consider a logical interface configured with a shaping rate of 10 Mbps and a guaranteed rate of 10 Mbps. At the queue level, parameters are configured as shown in Table 106 on page 743.

Table 106: Excess Priority Configuration

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Excess Rate	Traffic Sent To Queue
Q0	40%	NA	50%	10 Mbps
Q1	30%	NA	50%	10 Mbps
Q2	25%	NA	NA	0 Mbps
Q3	5%	NA	NA	1 Mbps

In this fifth example, Queue 0 is configured with an excess priority of **high** and all other queues have the default excess priority (**low**). Because there is no traffic on Queue 2, there is an excess bandwidth of 2.5 Mbps. Because Queue 0 has a higher excess priority, Queue 0 gets the entire excess bandwidth. So the expected output rates on the queues are 4 Mbps+ 2.5 Mbps= 6.5 Mbps for Queue 0, 3 Mbps for Queue 1, 0 Mbps for Queue 2, and 0.5 Mbps for Queue 3. Note that this behavior is different than regular priorities. With regular priorities, the transmission is still governed by transmit rates and the priority controls only the order in which the packets are picked up by the scheduler. So without excess configuration, if Queue 0 had a regular priority of **high** and there was 10 Mbps of traffic on all four queues, the traffic distribution would be 4 Mbps for Queue 0, 3 Mbps for Queue 1, 2.5 Mbps for Queue 2, and 0.5 Mbps for Queue 3 instead of giving all 10 Mbps to Queue 0. Excess priority traffic distributions are governed first by the excess priority and then by the excess rates. Also note that in this example, although the queues are in the excess region because they are transmitting above their configured transmit rates, the logical interface is still within its guaranteed rate. So at the logical interface level, the priority of the queues get promoted to a regular priority and this priority is used by the scheduler at the logical interface level.

The sixth and final example considers the effects of the default excess priority. When the excess priority for a queue is not configured explicitly, the excess priority is based on the regular priority. A regular priority of **high** maps to an excess priority of **high**. All other regular priorities map to an excess priority of **low**. When there is no regular priority configured, the regular and excess priorities are both set to **low**.

Copyright © 2019, Juniper Networks, Inc.

Configuring the Junos OS to Support Eight Queues on IQ Interfaces for T Series and M320 Routers

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

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

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

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

NOTE: The configuration at the **[edit class-of-service]** hierarchy level must also support eight queues per interface.

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

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

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

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

NOTE: Starting from Junos OS Release 14.1R8, 14.2R6, 15.1F6, 15.1R3, 15.1R4, and 16.1R1, the restricted queue PICs without the **max-queues-per-interface** configuration boot up with a maximum of eight queues per port and two operational ports (port 0 and 2). PICs with restricted queues include Quad T3 PIC, Quad E3 PIC, 4-port SONET/SDH OC3c/STM1 PIC, and 4-Port OC3 and 1-port OC12 PICs with SFP.

Copyright © 2019, Juniper Networks, Inc.

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

Related Documentation

- *Configuring the Junos OS to Support ILMI for Cell Relay Encapsulation on an ATM2 IQ PIC*
- *Configuring the Junos OS to Enable Larger Delay Buffers for T1, E1, and DS0 Interfaces Configured on Channelized IQ PICs*

BA Classifiers and ToS Translation Tables

On some PICs, the behavior aggregate (BA) translation tables are included for every logical interface (unit) protocol family configured on the logical interface. The proper default translation table is active even if you do not include any explicit translation tables. You can display the current translation table values with the **show class-of-service classifiers** command.

On Juniper Networks M40e, M120, M320 Multiservice Edge Routers, and T Series Core Routers with Enhanced IQ (IQE) PICs, or on any router or switch with IQ2 or Enhanced IQ2 (IQ2E) PICs, you can replace the type-of-service (ToS) bit value on the incoming packet header on a logical interface with a user-defined value. The new ToS value is used for all class-of-service processing and is applied before any other class-of-service or firewall treatment of the packet. The PIC uses the **translation-table** statement to determine the new ToS bit values.

You can configure a physical interface (port) or logical interface (unit) with up to three translation tables. For example, you can configure a port or unit with BA classification for IPv4 DSCP, IPv6 DSCP, and MPLS EXP. The number of frame relay data-link connection identifiers (DLCIs) (units) that you can configure on each PIC varies based on the number and type of BA classification tables configured on the interfaces.

Related Documentation

- Configuring ToS Translation Tables on page 745

Configuring ToS Translation Tables

On the IQE PICs, the behavior aggregate (BA) translation tables are included for every logical interface (unit) protocol family configured on the logical interface. The proper default translation table is active even if you do not include any explicit translation tables. You can display the current translation table values with the **show class-of-service classifiers** command.

On M40e, M120, M320, and T Series routers with IQE PICs, or on any device with IQ2 or Enhanced IQ2 PICs, you can replace the ToS bit value on the incoming packet header on a logical interface with a user-defined value. The new ToS value is used for all class-of-service processing and is applied before any other class-of-service or firewall

Copyright © 2019, Juniper Networks, Inc.

treatment of the packet. On the IQE PIC, the values configured with the **translation-table** statement determines the new ToS bit values.

Four types of translation tables are supported: IP precedence, IPv4 DSCP, IPv6 DSCP, and MPLS EXP. You can configure a maximum of eight tables for each supported type. If a translation table is enabled for a particular type of traffic, then behavior aggregate (BA) classification of the same type must be configured for that logical interface. In other words, if you configure an IPv4 translation table, you must configure IPv4 BA classification on the same logical interface.

The **from-code-points** statement establishes the values to match on the incoming packets. The **default** option is used to match all values not explicitly listed, and, as a single entry in the translation table, to mark all incoming packets on an interface the same way. The **to-code-point** statement establishes the target values for the translation. If an incoming packet header ToS bit configuration is not covered by the translation table list and a * option is not specified, the ToS bits in the incoming packet header are left unchanged.

You can define many translation tables, as long as they have distinct names. You apply a translation table to a logical interface at the **[edit class-of-service interfaces]** hierarchy level. Translation tables always translate "like to like." For example, a translation table applied to MPLS traffic can only translate from received EXP bit values to new EXP bit values. That is, translation tables cannot translate (for instance) from DSCP bits to INET precedence code points.

On the IQE PIC, incoming ToS bit translation is subject to the following rules:

- Locally generated traffic is not subject to translation.
- The **to-dscp-from-dscp** translation table type is not supported if an Internet precedence classifier is configured.
- The **to-inet-precedence-from-inet-precedence** translation table type is not supported if a DSCP classifier is configured.
- The **to-dscp-from-dscp** and **to-inet-precedence-from-inet-precedence** translation table types cannot be configured on the same unit.
- The **to-dscp-from-dscp** and **to-inet-precedence-from-inet-precedence** translation table types are supported for IPv4 packets.
- Only the **to-dscp-ipv6-from-dscp-ipv6** translation table type is supported for IPv6 packets.
- Only the **to-exp-from-exp** translation table type is supported for MPLS packets.

NOTE: Translation tables are not supported if fixed classification is configured on the logical interface.

A maximum of 32 distinct translation tables are supported on each IQE PIC. However, this maximum is limited by the number of classifiers configured along with translation tables because on the IQE PIC the hardware tables are not always merged. For example, if a translation table and a classifier are both configured on the same logical interface

Copyright © 2019, Juniper Networks, Inc.

(such as **unit 0**), there is only one hardware table and only one table added to the 32 translation table limit. However, if the translation table is configured on **unit 0** and the classifier on **unit 1** on the same physical interface, then two hardware tables are used and these two tables count toward the 32 maximum.

If you try to configure mutually exclusive translation tables on the same interface unit, you will get a warning message when you display or commit the configuration:

```
ge-0/1/1 {
    unit 0 {
        translation-table {
            ##
            ## Warning: to-dscp-from-dscp and
to-inet-precedence-from-inet-precedence not allowed on same unit
            ##
            to-inet-precedence-from-inet-precedence inet-trans-table;
            to-dscp-from-dscp dscp-trans-table;
        }
    }
}
```

ToS translation on the IQE PIC is a form of behavior aggregate (BA) classification. The IQE PIC does not support multifield classification of packets at the PIC level.

To configure ToS translation on the IQE PIC, include the **translation-table** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
translation-table {
  (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 | to-exp-from-exp |
    to-inet-precedence-from-inet-precedence) table-name {
    to-code-point value from-code-points (* | [ values ]);
  }
}
```

The following example procedure translates incoming DSCP values to the new values listed in the table. All incoming DSCP values other than **111111**, **111110**, **000111**, and **100111** are translated to **000111**:

1. Create and configure the translation table.

```
[edit class-of-service]
user@host# set translation-table to-dscp-from-dscp dscp-trans-table to-code-point
  000000 from-code-points 111111
user@host# set translation-table to-dscp-from-dscp dscp-trans-table to-code-point
  000001 from-code-points 111110
user@host# set translation-table to-dscp-from-dscp dscp-trans-table to-code-point
  111000 from-code-points [ 000111 100111 ]
user@host# set translation-table to-dscp-from-dscp dscp-trans-table to-code-point
  000111 from-code-points *
```

2. Apply the translation table to the logical interface input on the Enhanced IQ PIC

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service]
user@host# set interfaces so-1/0/0 unit 0 translation-table to-dscp-from-dscp
  dscp-trans
```

3. Verify the configuration.

   - To verify that the correct values are configured, use the **show class-of-service translation-table** command. The **show class-of-service translation-table** command displays the code points of all translation tables configured. All values are displayed, not just those configured:

     ```
     user@host> show class-of-service translation-table
     ```

     ```
     Translation Table: dscp-trans-table, Translation table type: dscp-to-dscp,
     Index: 6761
       From Code point    To Code Point
       000000                      000111
       000001                      000111
       000010                      000111
       000011                      000111
       000100                      000111
       000101                      000111
       000110                      000111
       000111                      111000
       001000                      000111
       001001                      000111
       001010                      000111
       001011                      000111
       001100                      000111
       001101                      000111
       001110                      000111
       001111                      000111
       010000                      000111
       010001                      000111
       010010                      000111
       010011                      000111
       010100                      000111
       010101                      000111
       010110                      000111
       010111                      000111
       011000                      000111
       011001                      000111
       011010                      000111
       011011                      000111
       011100                      000111
       011101                      000111
       011110                      000111
       011111                      000111
       100000                      000111
       100001                      000111
       100010                      000111
       100011                      000111
       100100                      000111
       100101                      000111
       100110                      000111
       100111                      111000
       101000                      000111
       101001                      000111
     ```


Copyright © 2019, Juniper Networks, Inc.

```
  101010                      000111
  101011                      000111
  101100                      000111
  101101                      000111
  101110                      000111
  101111                      000111
  110000                      000111
  110001                      000111
  110010                      000111
  110011                      000111
  110100                      000111
  110101                      000111
  110110                      000111
  110111                      000111
  111000                      000111
  111001                      000111
  111010                      000111
  111011                      000111
  111100                      000111
  111101                      000111
  111110                      000001
  111111                      000000
```

- To verify that the configured translation table is applied to the correct interface, use the **show class-of-service interface** ***interface-name*** command. The **show class-of-service interface** ***interface-name*** command displays the translation tables applied to the IQE interface:

  ```
  user@host> show class-of-service interface ge-0/1/1
  ```

  ```
  Physical interface: ge-0/1/1, Index: 156  From Code point    To Code Point
    Queues supported: 4, Queues in use: 4
      Scheduler map: <default>, Index: 2
      Chassis scheduler map: <default-chassis>, Index: 4

    Logical interface: so-2/3/0.0, Index: 68
      Object                  Name                    Type
  Index
      Rewrite                 exp-default             exp (mpls-any)
     29
      Classifier              dscp-default            dscp
      7
      Classifier              exp-default             exp
     10
      Translation Table       exp-trans-table         EXP_TO_EXP
  61925
  ```

4. Save the configuration.

   ```
   [edit]
   ```

```
user@host# commit
```

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 97

Copyright © 2019, Juniper Networks, Inc.

Configuring Hierarchical Layer 2 Policers on IQE PICs

The IQE PIC can police traffic at Layer 2 in a hierarchical manner. *Policing* is the practice of making sure that different streams of incoming traffic conform to certain parameters and limits. If the incoming traffic exceeds the established boundaries, that traffic can be marked or even ignored, depending on configuration. Hierarchical policing maintains two rates: an aggregate rate and a high-priority rate. The traffic is marked differently depending on service class (currently, the classes are expedited forwarding and nonexpedited forwarding). The expedited traffic has an additional rate configured, the guaranteed rate (CIR), which is only marked above that limit. If there is no expedited traffic present, then the non-expedited traffic is able to use the aggregate bandwidth rate before being marked with a packet loss priority. When expedited traffic is present, it is marked above the guaranteed rate, but also uses bandwidth from the nonexpedited range.

For example, consider an aggregate rate of 10 Mbps and a high-priority rate of 2 Mbps of a Fast Ethernet interface. The guaranteed rate is also set at 2 Mbps for expedited forwarding traffic. If there is no expedited traffic present, then nonexpedited traffic can use up to 10 Mbps before being marked. When expedited forwarding traffic is present, the expedited traffic is guaranteed 2 Mbps (of the 10 Mbps) without being marked, but is marked above the 2 Mbps limit. In this case, the nonexpedited forwarding traffic can use the remaining 8 Mbps before being marked.

Layer 2 policers configured on IQE PICs have the following limitations:

- Only one kind of policer is supported on a physical or logical interface. For example, a hierarchical or two- or three-color policer in the same direction on the same logical interface is not supported.
- Applying policers to both physical port and logical interface (policer chaining) is not supported.
- If there is no behavior aggregate classification, there is a limit of 64 policers per interface. (Usually, there will be a single policer per DLCI in frame relay and other logical interface types.)
- The policer should be independent of behavior aggregate classification. (Without a behavior aggregate, all traffic is treated as either expedited or non-expedited forwarding, depending on configuration.)
- With a behavior aggregate, traffic not matching any classification bits (such as DSCP or EXP) is policed as nonexpedited forwarding traffic.
- Only two levels of traffic policing are supported: **aggregate** and **premium**.

To configure Layer 2 policing on the IQE PIC, for each forwarding class:

1. Enable configuration of the forwarding classes.

```
[edit]
user@host# edit class-of-service forwarding-classes
```

Copyright © 2019, Juniper Networks, Inc.

2. Define the forwarding classes.

```
set class fc1 queue-num 0 priority high policing-priority premium
set class fc2 queue-num 1 priority low policing-priority normal
set class fc3 queue-num 2 priority low policing-priority normal
set class fc4 queue-num 3 priority low policing-priority normal
```

3. Configure the hierarchical policer.

   a. Enable configuration of the hierarchical policer.

```
[edit]
user@host# edit firewall hierarchical-policer hier_example1
```

   b. Configure the aggregate policer.

```
[edit firewall hierarchical-policer hier_example1 ]
user@host# set aggregate if-exceeding bandwidth-limit 70m burst-size-limit 1800
user@host# set aggregate then discard
```

   c. Configure the premium policer.

```
[edit firewall hierarchical-policer hier_example1 ]
user@host# set premium if-exceeding bandwidth-limit 70m burst-size-limit 3600
user@host# set premium then discard
```

4. Apply the policer to the logical on the IQE PIC.

```
[edit]
user@host# edit interfaces so-6/0/0 unit 0
user@host# set layer2-policer input-hierarchical-policer hier_example1
user@host# set family inet address 10.0.22.1/30
user@host# set family iso
user@host# set family mpls
```

Alternatively, to hierarchically rate-limit Layer 2 ingress traffic for all protocol families and for all logical interfaces configured on physical interface so-6/0/0, you could reference the policer from the physical interface configuration.

Related Documentation

- Controlling Network Access Using Traffic Policing Overview on page 115
- Configuring a Custom Forwarding Class for Each Queue on page 212

Copyright © 2019, Juniper Networks, Inc.

Configuring Excess Bandwidth Sharing on IQE PICs

Copyright © 2019, Juniper Networks, Inc.

The IQE PIC gives users more control over excess bandwidth sharing. You can set a shaping rate and a guaranteed rate on a queue or logical interface and control the excess bandwidth (if any) that can be used after all bandwidth guarantees have been satisfied. This section discusses the following topics related to excess bandwidth sharing on the IQE PIC:

On some types of PICs, including the IQ and IQ2, and Enhanced Queuing DPCs, you can configure either a committed information rate (CIR) using the **guaranteed-rate** statement or a peak information rate (PIR) using the **shaping-rate** statement. You can configure both a PIR and CIR, and in most cases the CIR is less than the value of PIR. For bursty traffic, the CIR represents the average rate of traffic per unit time and the PIR represents the maximum amount of traffic that can be transmitted in a given interval. In other words, the PIR (**shaping-rate**) establishes the maximum bandwidth available. The CIR (**guaranteed-rate**) establishes the minimum bandwidth available if all sources are active at the same time. Theoretically, the PIR or CIR can be established at the queue, logical interface, or physical interface level. In this section, the PIRs or CIRs apply at the queue or logical interface (or both) levels.

NOTE: You can configure a shaping rate at the physical interface, logical interface, or queue level. You can configure a guaranteed rate or excess rate only at the logical interface and queue level.

Once all of the bandwidth guarantees (the sum of the CIRs at that level) are met, there could still be some excess bandwidth available for use. In existing PICs, you have no control over how this excess bandwidth is used. For example, consider the situation shown in Table 107 on page 754 regarding a 10-Mbps physical interface. This example assumes that all queues are of the same priority. Also, if you do not specify a priority for the excess bandwidth, the excess priority is the same as the normal priority.

Table 107: Default Handling of Excess Traffic

Queue	Transmit Rate (CIR)	Shaping Rate (PIR)	Traffic Rate	Guaranteed Rate (Total = 6 Mbps)	Maximum Rate	Excess Bandwidth (Part of 4 Mbps Excess)	Expected Transmit Rate (Guarantee + Excess)
Q0	10%	80%	10 Mbps	1 Mbps	8 Mbps	0.73 Mbps	1.73 Mbps
Q1	20%	50%	10 Mbps	2 Mbps	5 Mbps	1.45 Mbps	3.45 Mbps
Q2	5%	5%	10 Mbps	0.5 Mbps	0.5 Mbps	0 Mbps	0.5 Mbps
Q3	25%	NA (“100%”)	10 Mbps	2.5 Mbps	10 Mbps	1.82 Mbps	4.32 Mbps

A 10-Mbps interface (the Traffic Rate column) has four queues, and the guaranteed rates are shown as percentages (Transmit Rate column) and in bits per second (Guaranteed Rate column). The table also shows the shaping rate (PIR) as a percentage (Shaping Rate column) and the actual maximum possible transmitted rate (Traffic Rate column)

Copyright © 2019, Juniper Networks, Inc.

on the oversubscribed interface. Note the guaranteed rates (CIRs) add up to 60 percent of the physical port speed or 6 Mbps. This means that there are 4 Mbps of "excess" bandwidth that can be used by the queues. This excess bandwidth is used as shown in the last two columns. One column (the Excess Bandwidth column) shows the bandwidth partitioned to each queue as a part of the 4-Mbps excess. The excess 4 Mbps bandwidth is shared in the ratio of the transmit rate (CIR) percentages of 10, 20, 5, and 25, adjusted for granularity. The last column shows the transmit rate the users can expect: the sum of the guaranteed rate plus the proportion of the excess bandwidth assigned to the queue.

Note that on PICs other than the IQE PICs the user has no control over the partitioning of the excess bandwidth. Excess bandwidth partitioning is automatic, simply assuming that the distribution and priorities of the excess bandwidth should be the same as the distribution and priorities of the other traffic. However, this might not always be the case and the user might want more control over excess bandwidth usage.

For more information on how excess bandwidth sharing is handled on the Enhanced Queuing DPC, see "Configuring Excess Bandwidth Sharing" on page 916.

On PICs other than IQE PICs, you can limit a queue's transmission rate by including the **transmit-rate** statement with the **exact** option at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level. However, on the IQE PIC, you can set a shaping rate independent of the transmit rate by including the **shaping-rate** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level. Also, other PICs share excess bandwidth (bandwidth left over once the guaranteed transmit rate is met) in an automatic, nonconfigurable fashion. You cannot configure the priority of the queues for the excess traffic on other PICs either.

To share excess bandwidth on IQE PICs, include the **excess-rate** statement along with the **guaranteed-rate** statement (to define the CIR) and the **shaping-rate** statement (to define the PIR):

```
[edit class-of-service traffic-control-profile profile-name]
[edit class-of-service schedulers scheduler-name]
excess-rate percent percentage;
guaranteed-rate (percent percentage | rate);
shaping-rate (percent percentage | rate);
```

To apply these limits to a logical interface, configure the statements at the **[edit class-of-service traffic-control-profile *profile-name*]** hierarchy level. To apply these limits to a specific queue, configure the statements at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level. You must also complete the configuration by applying the scheduler map or traffic control profile correctly.

You configure the excess rate as a percentage from 1 through 100. By default, excess bandwidth is automatically distributed as on other PIC types.

Copyright © 2019, Juniper Networks, Inc.

You can also configure a high or low priority for excess bandwidth by including the **excess-priority** statement with the **high** or **low** option at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level. This statement establishes the priority at the queue level, which then applies also at the logical and physical interface levels.

```
[edit class-of-service schedulers scheduler-name]
excess-priority (high | low);
```

NOTE: You cannot configure an excess rate for a logical interface if there is no guaranteed rate configured on any logical interface belonging to the physical interface.

The following example configures excess bandwidth sharing on logical interfaces of an IQE PIC by using twotraffic control profile:

1. Create the first traffic control profile called: for-unit-0-percent and specify the associated parameters for sharing the excess bandwidth.

   a. Specify a name for the traffic control profile to create it.

   ```
   [edit]
   user@host$ edit class-of-service traffic-control-profiles for-unit-0-percent
   ```

   b. Configure the maximum usage rate.

   ```
   [edit class-of-service traffic-control-profiles for-unit-0-percent]
   user@host$ set shaping-rate 10k
   ```

   c. Specify the guaranteed-rate (to define the CIR)

   ```
   [edit class-of-service traffic-control-profiles for-unit-0-percent]
   user@host$ guaranteed-rate 1k
   ```

   d. Specify the excess-rate

   ```
   [edit class-of-service traffic-control-profiles for-unit-0-percent]
   user@host$ excess-rate percent 30
   ```

2. Create the second traffic control profile called: for-unit-1-proportion and specify the associated parameters for sharing the excess bandwidth.

   a. Specify a name for the traffic control profile to create it.

   ```
   [edit]
   user@host$ edit class-of-service traffic-control-profiles for-unit-1-proportion
   ```

   b. Configure the maximum usage rate.

   ```
   [edit class-of-service traffic-control-profiles for-unit-1-proportion]
   user@host$ set shaping-rate 5m
   ```

   c. Specify the percentage or proportion of excess bandwidth traffic to share.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service traffic-control-profiles for-unit-1-proportion]
user@host$ excess-rate percent 30
```

d. Specify the priority for excess bandwidth.

3. Verify the configuration.

```
user@host> show class-of-service traffic-control-profile
for-unit-0-percent {
    shaping-rate 10k;
    guaranteed-rate 1k;
    excess-rate percent 30;
}
for-unit-1-proportion {
    shaping-rate 20k;
    guaranteed-rate 10k;
    excess-rate percent 35;
}
```

The following example configures the excess rate in a scheduler:

1. Create the first scheduler called: scheduler-for-excess-low and specify the associated parameters for sharing the excess bandwidth.

   a. Specify a name for the scheduler to create it.

   ```
   [edit]
   user@host$ edit class-of-service schedulers scheduler-for-excess-low
   ```

   b. Specify the transmit rate for the scheduler.

   ```
   [edit class-of-service schedulers scheduler-for-excess-low]
   user@host$ set transmit-rate 1m
   ```

   c. Configure the maximum usage rate.

   ```
   [edit class-of-service schedulers scheduler-for-excess-low]
   user@host$ set shaping-rate 5m
   ```

   d. Specify the percentage or proportion of excess bandwidth traffic to share.

   ```
   [edit class-of-service schedulers scheduler-for-excess-low]
   user@host$ excess-rate percent 30
   ```

   e. Specify the priority of excess bandwidth traffic on the scheduler.

   ```
   [edit class-of-service schedulers scheduler-for-excess-low]
   user@host$ excess-priority low
   ```

2. Create the second scheduler called: scheduler-for-excess-high and specify the associated parameters for sharing the excess bandwidth.

   a. Specify a name for the traffic control profile to create it.

Copyright © 2019, Juniper Networks, Inc.

```
[edit]
user@host$ edit class-of-service schedulers scheduler-for-excess-high
```

b. Specify the transmit rate for the scheduler.

```
[edit class-of-service schedulers scheduler-for-excess-low]
user@host$ set transmit-rate percent 20
```

c. Configure the maximum usage rate.

```
[edit class-of-service schedulers scheduler-for-excess-high]
user@host$ set shaping-rate percent 30
```

d. Specify the percentage or proportion of excess bandwidth traffic to share.

```
[edit class-of-service schedulers scheduler-for-excess-high]
user@host$ excess-rate percent 25
```

e. Specify the priority of excess bandwidth traffic on the scheduler.

```
[edit class-of-service schedulers scheduler-for-excess-high]
user@host$ excess-priority high
```

3. Verify the configuration.

```
user@host> show class-of-service schedulers
scheduler-for-excess-low {
    transmit-rate 1m;
    shaping-rate 5m;
    excess-rate percent 30;
    excess-priority low;
}
scheduler-for-excess-high {
    transmit-rate percent 20;
    shaping-rate percent 30;
    excess-rate percent 25;
    excess-priority high;
}
```

NOTE: All of these parameters apply to egress traffic only and only for per-unit schedulers. That is, there is no hierarchical or shared scheduler support.

Related Documentation

- Configuring Schedulers on page 258
- Excess Rate and Excess Priority Configuration Examples on page 290

Copyright © 2019, Juniper Networks, Inc.

Configuring Rate-Limiting Policers for High Priority Low-Latency Queues on IQE PICs

You can rate-limit the strict-high and high queues on the IQE PIC. Without this limiting, traffic that requires low latency (delay) such as voice can block the transmission of medium-priority and low-priority packets. Unless limited, high and strict-high traffic is always sent before lower priority traffic, causing the lower priority queues to "starve" and cause timeouts and unnecessarily resent packets.

On the IQE PIC you can rate-limit queues before the packets are queued for output. All packets exceeding the configured rate limit are dropped, so care is required when establishing this limit. This model is also supported on IQ2 PICs and is the only way to perform egress policing on IQE PICs. This feature introduces no new configuration statements.

Although intended for low-latency traffic classes such as voice, the configuration allows any queue to be rate-limited. However, the configuration requires the rate-limited queue to have either a high or strict-high priority.

NOTE: You can configure a low-latency static policer for only one rate-limited queue per scheduler map. You can configure up to 1024 low-latency static policers.

This example limits the transmit rate of a strict-high expedited-forwarding queue to 1 Mbps. The scheduler and scheduler map are defined, and then applied to the traffic at the **[edit interfaces]** and **[edit class-of-service]** hierarchy levels:

1. Define the scheduler:
   a. Specify a name for the scheduler to create it.

      ```
      [edit]
      user@host# edit class-of-service schedulers scheduler-1
      ```

   b. Specify the transmit rate.

      ```
      [edit class-of-service schedulers scheduler-1]
      user@host# set transmit-rate 1m rate-limit
      ```

   c. Specify the priority of the scheduler.

      ```
      [edit class-of-service schedulers scheduler-1]
      user@host# set priority strict-high
      ```

2. Define the scheduler map:
   a. Specify a name for the scheduler map to create it.

      ```
      [edit]
      user@host# edit class-of-service scheduler-maps scheduler-map1
      ```

Copyright © 2019, Juniper Networks, Inc.

b. Map the EF forwarding class to the scheduler.

```
[edit class-of-service scheduler-maps scheduler-map-1]
user@host# set forwarding-class expedited-forwarding scheduler scheduler-1
```

3. Configure the physical interface.

   This example uses Frame Relay encapsulation and enables per-unit scheduling, which enables you to apply scheduling to the Frame Relay DLCI.

   a. Specify the physical interface of the interface.

```
[edit]
user@host# edit interfaces so-2/0/0
```

   b. Enable the association of scheduler map names with logical interfaces.

```
[edit interfaces s0-2/0/0]
user@host# set per-unit-scheduler
```

   c. Specify the encapsulation type.

```
[edit interfaces s0-2/0/0]
user@host# set encapsulation frame-relay
```

4. Configure the logical interface and specify the Frame Relay DLCI.

   a. Specify the logical interface number.

```
[edit interfaces s0-2/0/0]
user@host# edit unit 0
```

   b. Specify the data-link connection identifier (DLCI).

```
[edit interfaces s0-2/0/0 unit 0]
user@host# set dlci 1
```

5. Apply the scheduler map to the logical interface:

   a. Specify the physical and logical interface to which you want to apply the scheduler map.

```
[edit]
user@host# edit class-of-service interfaces so-2/0/0 unit 0
```

   b. Specify the name of the scheduler map you created.

```
[edit class-of-service interfaces so-2/0/0 unit 0]
user@host# set scheduler-map scheduler-map1
```

   c. Specify the amount of bandwidth to be allocated for the logical interface.

```
[edit class-of-service interfaces so-2/0/0 unit 0]
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set shaping-rate 2m
```

6. You can issue the following operational mode commands to verify your configuration (the first shows the rate limit in effect):

   - **show class-of-service scheduler-map scheduler-map scheduler-map-1**
   - **show class-of-service interface so-2/0/0**

```
[edit class-of-service]
schedulers {
    scheduler-1 {
        transmit-rate 1m rate-limit;
        priority strict-high;
    }
}
scheduler-maps {
    scheduler-map1 {
        forwarding-class expedited-forwarding scheduler scheduler-1;
    }
}
[edit interfaces]
s0-2/0/0 {
    per-unit-scheduler;
    encapsulation frame-relay;
    unit 0 {
      dlci 1;
    }
}
[edit class-of-service]
interfaces {
    so-2/0/0 {
      unit 0 {
          scheduler-map scheduler-map1;
          shaping-rate 2m;
          }
      }
}
```

Related Documentation

- Configuring Schedulers on page 258
- Configuring Scheduler Maps on page 258
- show class-of-service scheduler-map on page 1406

Applying Scheduler Maps and Shaping Rate to Physical Interfaces on IQ PICs

This topic describes how to configure and apply scheduler maps and shaping rates to physical interfaces on various types of IQ PICs.

You can specify a peak bandwidth rate (shaping rate) in bps, 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). For physical interfaces, the range is from 1000 through 6,400,000,000,000 bps. For the IQ2 Gigabit Ethernet PIC, the minimum is

Copyright © 2019, Juniper Networks, Inc.

80,000 bps, and for the IQ2 10 Gigabit Ethernet PIC, the minimum is 160,000 bps. (For logical interfaces, the range is 1000 through 32,000,000,000 bps.) The sum of the bandwidths you allocate to all physical interfaces on a PIC must not exceed the bandwidth of the PIC.

NOTE: For MX Series routers, the shaping rate value for the physical interface at the **[edit class-of-service interfaces *interface-name*]** hierarchy level must be a minimum of 160 Kbps.

If you configure a shaping rate that exceeds the physical interface bandwidth, the new configuration is ignored, and the previous configuration remains in effect. For example, if you configure a shaping rate that is 80 percent of the physical interface bandwidth, then change the configuration to 120 percent of the physical interface bandwidth, the 80 percent setting remains in effect. This holds true unless the PIC is restarted, in which case the default bandwidth goes into effect. As stated previously, the default bandwidth is based on the channel bandwidth and the time slot allocation.

Optionally, you can instead configure scheduling and rate shaping on logical interfaces, as described in "Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs" on page 300. In general, logical and physical interface traffic shaping is mutually exclusive. You can include the **shaping-rate** statement at the **[edit class-of-service interfaces *interface-name*]** hierarchy level or the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level, but not both. For Gigabit Ethernet IQ2 and IQ2E PICs, you can configure hierarchical traffic shaping, meaning the shaping is performed on both the physical interface and the logical interface. For more information, see "Configuring Input Shaping Rates for Both Physical and Logical Interfaces" on page 326.

For more information, see the following sections:

- Examples: Applying a Shaping Rate on page 762
- Examples: Applying a Scheduler Map and Shaping Rate to Physical Interfaces on IQ PICs on page 766

Examples: Applying a Shaping Rate

This topic shows you how to calculate shaping rates and provides two examples of how to configure and apply a shaping rate. It includes the following sections:

- Shaping Rate Calculations on page 762
- Example: Applying a Shaping Rate to a Clear-Channel T1 Interface on a Channelized T1 IQ PIC on page 764
- Example: Applying a Shaping Rate to a Clear-Channel E1 Interface on a Channelized E1 IQ PIC on page 765

Shaping Rate Calculations

For shaping rate and WRR, the information included in the calculations varies by PIC type, as shown in Table 108 on page 763.

Copyright © 2019, Juniper Networks, Inc.

NOTE: The 10-port 10-Gigabit Oversubscribed Ethernet (OSE) PICs and Gigabit Ethernet IQ2 PICs are unique in supporting ingress scheduling and shaping. The calculations shown for 10-port 10-Gigabit OSE and Gigabit Ethernet IQ2 PICs apply to both ingress and egress scheduling and shaping. For other PICs, the calculations apply to egress scheduling and shaping only.

For more information, see "CoS on Enhanced IQ2 PICs Overview" on page 794.

Table 108: Shaping Rate and WRR Calculations by PIC Type

PIC Type	Platform	Shaping Rate and WRR Calculations Include
10-port 10-Gigabit OSE PIC	T Series Core Routers	For ingress and egress: L3 header + L2 header + frame check sequence (FCS) + interpacket gap (IPG) + preamble
Gigabit Ethernet IQ2 PIC	All	For ingress and egress: L3 header + L2 header + frame check sequence (FCS)
Gigabit Ethernet IQ PIC	All	L3 header + L2 header + FCS
IQ PIC with a SONET/SDH interface	All	L3 header+ L2 header + FCS

Copyright © 2019, Juniper Networks, Inc.

Example: Applying a Shaping Rate to a Clear-Channel T1 Interface on a Channelized T1 IQ PIC

To apply a shaping rate to a clear-channel T1 interface on a channelized T1 IQ PIC:

1. Configure the physical and logical interfaces.

   a. Specify the physical interface to configure.

   ```
   [edit]
   user@host# edit interfaces ct1-2/1/0
   ```

   b. Configure the channelized T1 IQ PIC as unpartitioned, clear channel.

   ```
   [edit interfaces ct1-2/1/0]
   user@host# set no-partition interface-type t1
   ```

   c. Specify the logical interface to configure.

   ```
   [edit interfaces]
   user@host# edit t1-2/1/0 unit 0
   ```

   d. Specify the IPv4 family and IP address

   ```
   [edit interfaces ]
   user@host# set family inet address 10.40.1.1/30
   ```

2. Configure traffic shaping by specifying the amount of bandwidth to be allocated to the logical interface.

   a. Specify the interface to configure.

   ```
   [edit]
   user@host# edit class-of-service interfaces t1-2/1/0
   ```

   b. Specify the shaping rate for the interface.

   ```
   [edit class-of-service interfaces t1-2/1/0]
   user@host# set shaping-rate 3000
   ```

3. Verify the configuration.

```
[edit interfaces]
```

```
user@ host# show
```

```
ct1-2/1/0 {
    no-partition interface-type t1;
}
t1-2/1/0 {
    unit 0 {
        family inet {
            address 10.40.1.1/30;
        }
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
}
```

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

```
user@ host# show
```

```
t1-2/1/0 {
    shaping-rate 160k;
}
```

4. Save your configuration.

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

Example: Applying a Shaping Rate to a Clear-Channel E1 Interface on a Channelized E1 IQ PIC

To apply a shaping rate to a clear-channel E1 interface on a channelized E1 IQ PIC:

1. Configure the physical and logical interfaces.
   a. Specify the physical interface to configure.

```
[edit]
user@host# edit interfaces ce1-2/1/0
```

   b. Configure the channelized T1 IQ PIC as unpartitioned, clear channel.

```
[edit interfaces ce1-2/1/0]
user@host# set no-partition interface-type e1
```

   c. Specify the logical interface to configure.

```
[edit interfaces]
user@host# edit e1-2/1/0 unit 0
```

   d. Specify the IPv4 family and IP address

```
[edit interfaces ]
user@host# set family inet address 10.40.1.1/30
```

2. Configure traffic shaping by specifying the amount of bandwidth to be allocated to the logical interface.
   a. Specify the interface to configure.

```
[edit]
user@host# edit class-of-service interfaces e1-2/1/0
user@host#
```

   b. Specify the shaping rate for the interface.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces e1-2/1/0]
user@host# set shaping-rate 4000
```

3. Verify the configuration.

```
[edit interfaces]
```

```
user@ host# show
```

```
ce1-2/1/0 {
    no-partition interface-type e1;
}
e1-2/1/0 unit 0 {
    unit 0 {
        family inet {
            address 10.40.1.1/30;
        }
    }
}
```

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

```
user@ host# show
```

```
e1-2/1/0 {
    shaping-rate 160k;
}
```

4. Save your configuration.

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

Examples: Applying a Scheduler Map and Shaping Rate to Physical Interfaces on IQ PICs

The following sections provide examples of how to configure and apply a scheduler map and shaping rate to various physical interface types.

The following examples are included:

- Example: Applying a Scheduler Map and Shaping Rate to a DS0 Channel of a Channelized T1 Interface on a Channelized T1 IQ PIC on page 767
- Example: Applying a Scheduler Map and Shaping Rate to DS0 Channels of a Channelized E1 Interface on a Channelized E1 IQ PIC on page 769
- Applying a Scheduler Map and Shaping Rate to a Clear-Channel T3 Interface on a Channelized DS3 IQ PIC on page 772
- Applying a Scheduler Map and Shaping Rate to Fractional T1 Interfaces on a Channelized DS3 IQ PIC on page 774
- Applying a Scheduler Map and Shaping Rate to a DS0 Channel of a T1 Interface in a Channelized T3 Interface on a Channelized DS3 IQ PIC on page 777

Copyright © 2019, Juniper Networks, Inc.

Example: Applying a Scheduler Map and Shaping Rate to a DS0 Channel of a Channelized T1 Interface on a Channelized T1 IQ PIC

To apply a scheduler map and shaping rate to a clear-channel T1 interface on a channelized T1 IQ PIC:

For this procedure, you must also configure a scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

1. Configure the physica interface.
   a. Specify the physical interface to configure.

      ```
      [edit]
      user@host# edit interfaces ct1-0/0/9
      ```

   b. Configure the channelized T1 IQ PIC as unpartitioned, clear channel.

      ```
      [edit interfaces ct1-2/1/0]
      user@host# set partition 1 timeslots 1-2 interface-type ds
      ```

2. Configure the logical interface.
   a. Specify the logical interface (DS0) to configure.

      ```
      [edit interfaces]
      user@host# edit ds-0/0/9:1
      ```

   b. Disable the sending of keepalives on the interface.

      ```
      [edit interfaces ds-0/0/0:1]
      set no-keepalives
      ```

   c. Specify the IPv4 family and IP address for the logical interface.

      ```
      [edit interfaces ds-0/0/0:1]
      user@host# set unit 0 family inet address 10.10.1.1/30
      ```

3. Apply the scheduler map and shaping rate to the interface.

   NOTE: Be sure you have previously configured the scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

   a. Specify the interface to configure.

      ```
      [edit]
      user@host# edit class-of-service interfaces ds-0/0/9:1
      ```

   b. Specify the name of the scheduler map to apply to the interface.

      ```
      [edit class-of-service interfaces ds-0/0/9:1]
      ```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set scheduler-map sched_port_1
```

c. Specify the amount of bandwidth to allocate to the interface.

```
set shaping-rate 2000
```

4. Verify the configuration.

```
[edit interfaces]
```

```
user@ host# show
```

```
ct1-0/0/9 {
    partition 1 timeslots 1-2 interface-type ds;
}
ds-0/0/9:1 {
    no-keepalives;
    unit 0 {
        family inet {
            address 10.10.1.1/30;
        }
    }
}
```

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

```
user@ host# show
```

```
ds-0/0/0:1 {
    scheduler-map sched_port_1;
    shaping-rate 160k;
}
```

5. Save your configuration.

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


Copyright © 2019, Juniper Networks, Inc.

Example: Applying a Scheduler Map and Shaping Rate to DS0 Channels of a Channelized E1 Interface on a Channelized E1 IQ PIC

To apply a scheduler map and shaping rate to a clear-channel E1 interface on a channelized E1 IQ PIC:

For this procedure, you must also configure a scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

1. Configure the physical.
   a. Specify the physical interface to configure.

      ```
      [edit]
      user@host# edit interfaces ce1-1/3/1
      ```

   b. Configure the channelized E1 IQ PIC as unpartitioned, clear channel.

      ```
      [edit interfaces ce1-1/3/1]
      user@host# set partition 1 timeslots 1-4 interface-type ds
      user@host# set partition 2timeslots 5–6 interface-type ds
      ```

2. Configure the first logical interface (DS0).
   a. Specify the logical interface to configure.

      ```
      [edit interfaces]
      user@host# edit ds-1/3/1:1
      ```

   b. Disable the sending of keepalives on the interface.

      ```
      [edit interfaces ds-1/3/1:1]
      set no-keepalives
      ```

   c. Specify the IPv4 family and IP address for the logical interface.

      ```
      [edit interfaces ds-1/3/1:1]
      user@host# set unit 0 family inet address 10.10.1.1/30
      ```

3. Configure the second logical interface (DS0).
   a. Specify the logical interface to configure.

      ```
      [edit interfaces]
      user@host# edit ds-1/3/1:2
      ```

   b. Disable the sending of keepalives on the interface.

      ```
      [edit interfaces ds-1/3/1:2]
      set no-keepalives
      ```

   c. Specify the IPv4 family and IP address for the logical interface.

      ```
      [edit interfaces ds-1/3/1:2]
      ```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set unit 0 family inet address 10.10.1.5/30
```

4. Apply the scheduler map and shaping rate to the first logical interface.

   NOTE: Be sure you have previously configured the scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

   a. Specify the logical interface to configure.

   ```
   [edit]
   user@host# edit class-of-service interfaces ds-1/3/1:1
   ```

   b. Specify the name of the scheduler map to apply to the logical interface.

   ```
   [edit class-of-service interfaces ds-1/3/1:1]
   user@host# set scheduler-map sched_port_1
   ```

   c. Specify the amount of bandwidth to allocate to the interface.

   ```
   [edit class-of-service interfaces ds-1/3/1:1]
   set shaping-rate 1000
   ```

5. Apply the scheduler map and shaping rate to the second logical interface.

   NOTE: Be sure you have previously configured the scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

   a. Specify the logical interface on which you want to apply the scheduler.

   ```
   [edit]
   user@host# edit class-of-service interfaces ds-1/3/1:2
   ```

   b. Specify the name of the scheduler map to apply to the interface.

   ```
   [edit class-of-service interfaces ds-1/3/1:2]
   user@host# set scheduler-map sched_port_1
   ```

   c. Specify the amount of bandwidth to allocate to the interface.

   ```
   [edit class-of-service interfaces ds-1/3/1:2]
   set shaping-rate 1500
   ```

6. Verify the configuration.

   ```
   [edit interfaces]
   user@ host# show
   ```


Copyright © 2019, Juniper Networks, Inc.

```
ce1-1/3/1 {
    partition 1 timeslots 1-4 interface-type ds;
    partition 2 timeslots 5-6 interface-type ds;
}
ds-1/3/1:1 {
    no-keepalives;
    unit 0 {
        family inet {
            address 10.10.1.1/30;
        }
    }
}
ds-1/3/1:2 {
    no-keepalives;
    unit 0 {
        family inet {
            address 10.10.1.5/30;
        }
    }
}
```

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

```
user@ host# show
```

```
interfaces {
  ds-1/3/1:1 {
    scheduler-map sched_port_1;
    shaping-rate 1000;
  }
  ds-1/3/1:2 {
    scheduler-map sched_port_1;
    shaping-rate 1500;
  }
}
```

7. Save your configuration.

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

Copyright © 2019, Juniper Networks, Inc.

Applying a Scheduler Map and Shaping Rate to a Clear-Channel T3 Interface on a Channelized DS3 IQ PIC

To apply a scheduler map and shaping rate to a clear-channel T3 interface on a channelized DS3 IQ PIC:

For this procedure, you must also configure a scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

1. Configure the channelized T3 IQ interface.
   a. Specify the physical interface to configure.

      ```
      [edit]
      user@host# edit interfaces ct3-2/1/0
      ```

   b. Configure the interface as unpartitioned.

      ```
      user@host# set no-partition
      ```

2. Configure the channelized DS3 interface.
   a. Specify the interface name.

      ```
      [edit interface]
      user@host# edit t3-2/1/0
      ```

   b. Specify the IPv4 family and IP address for the logical interface.

      ```
      [edit interfaces t3-2/1/0]
      user@host# set unit 0 family inet address 10.40.1.1/30
      ```

3. Apply the scheduler map and shaping rate on the channelized DS3 interface.

   NOTE: Be sure you have previously configured the scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

   a. Specify the interface name.

      ```
      [edit]
      user@host# edit class-of-service interfaces t3-2/1/0
      ```

   b. Specify the name of the scheduler map to apply to the interface.

      ```
      [edit class-of-service interfaces t3-2/1/0]
      user@host# set scheduler-map sched_port_1
      ```

   c. Specify the amount of bandwidth to allocate to the interface.

      ```
      [edit class-of-service interfaces t3-2/1/0]
      set shaping-rate 2500
      ```


Copyright © 2019, Juniper Networks, Inc.

4. Verify the configuration.

```
[edit interfaces]
```

```
user@ host# show
```

```
ct3-2/1/0 {
  no-partition;
}
t3-2/1/0 {
   unit 0 {
     family inet {
       address 10.40.1.1/30;
     }
   }
}
```

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

```
user@ host# show
```

```
interfaces {
  t3-2/1/0 {
    shaping-rate 2500;
    unit 0 {
      scheduler-map sched_port_1;
    }
  }
}
```

5. Save your configuration.

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

Copyright © 2019, Juniper Networks, Inc.

Applying a Scheduler Map and Shaping Rate to Fractional T1 Interfaces on a Channelized DS3 IQ PIC

To apply a scheduler map and shaping rate to a fractional T1 interfaces on a channelized DS3 IQ PIC:

For this procedure, you must also configure a scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

1. Configure the physical interface.
   a. Specify the name of the physical interface.

      ```
      [edit]
      user@host# edit interfaces ct3-1/1/3
      ```

   b. Configure the interface as a partitioned T1 interface.

      ```
      [edit interfaces ct3-1/1/3]
      user@host# set partition 1-3 interface-type t1
      ```

2. Configure the first logical interface (T1 interface).
   a. Specify the logical interface name.

      ```
      [edit interfaces]
      user@host# edit t1-1/1/3:1
      ```

   b. Configure T1-specific physical interface properties.

      ```
      [edit interfaces t1-1/1/3:1]
      set t1-options timeslots 1-2
      ```

   c. Specify the IPv4 family and IP address for the logical interface.

      ```
      [edit interfaces t1-1/1/3:1]
      user@host# set unit 0 family inet address 10.10.1.1/30
      ```

3. Configure the second logical interface (T1 interface).
   a. Specify the logical interface name.

      ```
      [edit interfaces]
      user@host# edit t1-1/1/3:2
      ```

   b. Configure T1-specific physical interface properties.

      ```
      [edit interfaces t1-1/1/3:2]
      set t1-options timeslots 3–6
      ```

   c. Specify the IPv4 family and IP address for the logical interface.

      ```
      [edit interfaces t1-1/1/3:2]
      user@host# set unit 0 family inet address 10.10.1.5/30
      ```


Copyright © 2019, Juniper Networks, Inc.

4. Configure the third logical interface (T1 interface).

   a. Specify the logical interface name.

   ```
   [edit interfaces]
   user@host# edit t1-1/1/3:3
   ```

   b. Configure T1-specific physical interface properties.

   ```
   [edit interfaces t1-1/1/3:3]
   set t1-options timeslots 7–12
   ```

   c. Specify the IPv4 family and IP address for the logical interface.

   ```
   [edit interfaces t1-1/1/3:3]
   user@host# set unit 0 family inet address 10.10.1.9/30
   ```

5. Apply the scheduler map and shaping rate to the T1 logical interfaces.

   NOTE: Be sure you have previously configured the scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

   a. Configure the first interface by specifying the interface name, appying the scheduler map to the interface, and specifying the amount of bandwidth to allocate to the interface.

   ```
   [edit]
   user@host# edit class-of-service
   user@host# set interfaces t1-1/1/3:1 scheduler-map sched_port_1 shaping-rate 1200
   ```

   b. Configure the second interface by specifying the interface name, appying the scheduler map to the interface, and specifying the amount of bandwidth to allocate to the interface.

   ```
   [edit class-of-service]
   user@host# edit interfaces t1-1/1/3:2 scheduler-map sched_port_1 shaping-rate
     1300
   ```

   c. Configure the third interface by specifying the interface name, appying the scheduler map to the interface, and specifying the amount of bandwidth to allocate to the interface.

   ```
   [edit class-of-service]
   user@host# edit interfaces t1-1/1/3:3 scheduler-map sched_port_1 shaping-rate
     1400
   ```

6. Verify the configuration.

   ```
   [edit interfaces]

   user@ host# show
   ```

Copyright © 2019, Juniper Networks, Inc.

```
ct3-1/1/3 {
    partition 1-3 interface-type t1;
}
t1-1/1/3:1 {
    t1-options {
        timeslots 1-2;
    }
    unit 0 {
        family inet {
            address 10.10.1.5/30;
        }
    }
}
t1-1/1/3:2 {
    t1-options {
        timeslots 3-6;
    }
    unit 0 {
        family inet {
            address 10.10.1.5/30;
        }
    }
}
t1-1/1/3:3 {
    t1-options {
        timeslots 7-12;
    }
    unit 0 {
        family inet {
            address 10.10.1.9/30;
        }
    }
}
```

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

```
user@ host# show
```

```
t1-1/1/3:1 {
  scheduler-map sched_port_1;
  shaping-rate 1200;
}
t1-1/1/3:2 {
  scheduler-map sched_port_1;
  shaping-rate 1300;
}
t1-1/1/3:3 {
  scheduler-map sched_port_1;
  shaping-rate 1400;
}
}
```

7. Save your configuration.

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


Copyright © 2019, Juniper Networks, Inc.

Applying a Scheduler Map and Shaping Rate to a DS0 Channel of a T1 Interface in a Channelized T3 Interface on a Channelized DS3 IQ PIC

To apply a scheduler map and shaping rate to a DS0 Channel of a T1 Interface in a Channelized T3 Channelized DS3 IQ PIC:

For this procedure, you must also configure a scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

1. Configure the physical interface.

   a. Specify the name of the physical interface.

   ```
   [edit]
   user@host# edit interfaces ct1-2/1/3
   ```

   b. Configure the interface as a partitioned, specify the timeslots for the interface, and specify the interface type.

   ```
   [edit interfaces ct3-2/1/3]
   user@host# set partition 1 timeslots 1-4 interface-type ds
   ```

2. Configure the DS0 interface.

   ```
   [edit interfaces]
   user@host# set ds-2/1/3:1:1 unit 0 family inet address 10.20.144.1/30
   ```

3. Apply the scheduler map and shaping rate to the logical interface.

   NOTE: Be sure you have previously configured the scheduler map. For details on configuring the scheduler map, see "Configuring Scheduler Maps" on page 258.

   a. Configure the interface by specifying the interface name, applying the scheduler map to the interface, and specifying the amount of bandwidth to allocate to the interface.

   ```
   [edit class-of-service]
   user@host# set interfaces ds-2/1/3:1:1 scheduler-map sched_port_1 shaping-rate
     1100
   ```

4. Verify the configuration.

   ```
   [edit interfaces]
   ```

   ```
   user@ host# show
   ```

   ```
   ct3-2/1/3 {
   partition 1 interface-type ct1;
   }
   ```

Copyright © 2019, Juniper Networks, Inc.

```
ct1-2/1/3:1 {
  partition 1 timeslots 1-4 interface-type ds;
}
ds-2/1/3:1:1 {
  unit 0 {
    family inet {
      address 10.20.144.1/30;
    }
  }
}
```

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

```
user@ host# show
```

```
interfaces {
  ds-2/1/3:1:1 {
    scheduler-map sched_port_1;
    shaping-rate 1100;
  }
}
```

5. Save your configuration.

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

Related Documentation

- Applying Scheduler Maps to Physical Interfaces on page 259
- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300

Applying Scheduler Maps to Chassis-Level Queues

On Intelligent Queuing (IQ) and Intelligent Queuing 2 (IQ2) interfaces, as well as on the 10x10GE MIC with SFP+, the traffic that is fed from the packet forwarding components into the PIC uses low packet loss priority (PLP) by default and is distributed evenly across the four chassis queues (not PIC queues), regardless of the scheduling configuration for each logical interface. This default behavior can cause traffic congestion.

The default chassis scheduler allocates resources for queue 0 through queue 3, with 25 percent of the bandwidth allocated to each queue. When you configure the chassis to use more than four queues, you must configure and apply a custom chassis scheduler to override the default chassis scheduler.

To apply a custom chassis scheduler:

1. Specify the interface on which to apply the scheduler.

```
[edit]
user@host# edit class-of-service interfaces interface-name
```


Copyright © 2019, Juniper Networks, Inc.

For example:

```
[edit]
user@host# edit class-of-service interfaces so-0/1/*
```

2. Specify the name of the custom scheduler you want to apply to the interface.

```
[edit class-of-service interfaces so-0/1/0]
user@host# set scheduler-map-chassis map-name
```

To control the aggregated traffic transmitted from the chassis queues into the PIC, you can configure the chassis queues to derive their scheduling configuration from the associated logical interface's.

To configure the chassis queues to derive their scheduling from the associated logical interfaces:

1. Specify the logical interfaces from which to derive the scheduling configuration.

```
[edit]
user@host# edit class-of-service interfaces interface-name
```

For example:

```
[edit]
user@host# edit class-of-service interfaces so-0/1/*
```

2. Specify that the scheduler configuration is derived from the specified logical interfaces.

```
[edit class-of-service interfaces so-0/1/0]
user@host# set scheduler-map-chassis derived
```

CAUTION: If you specify the **scheduler-map-chassis derived** statement in the configuration, packet loss might occur when you subsequently add or remove logical interfaces at the **[edit interfaces *interface-name*]** hierarchy level.

When fragmentation occurs on the egress interface, the first set of packet counters displayed in the output of the **show interfaces queue** command show the post-fragmentation values. The second set of packet counters (under the **Packet Forwarding Engine Chassis Queues** field) show the pre-fragmentation values. For more information about the **show interfaces queue** command, see the CLI Explorer.

You can specify both the **scheduler-map** and the **scheduler-map-chassis derived** statements in the same interface configuration. The **scheduler-map** statement controls the scheduler inside the PIC, while the **scheduler-map-chassis derived** statement controls the aggregated traffic transmitted into the entire PIC.

Copyright © 2019, Juniper Networks, Inc.

NOTE: For the Gigabit Ethernet IQ PIC, you must specify both the **scheduler-map** and the **scheduler-map-chassis derived** statements in the interface configuration.

Generally, when you specify the **scheduler-map-chassis** statement in the configuration, you must use an interface wildcard for the interface name, as in ***type-fpc/pic/****. The wildcard must use this format—for example. **so-1/2/***, which means all interfaces on FPC slot 1, PIC slot 2. There is one exception—you can apply the chassis scheduler map to a specific interface on the Gigabit Ethernet IQ PIC only.

According to Junos OS wildcard rules, specific interface configurations override wildcard configurations. For chassis scheduler map configurations, this rule does not apply; instead, specific interface CoS configurations are added to the chassis scheduler map configuration. For more information about how wildcards work with chassis scheduler maps, see "Examples: Scheduling Packet Forwarding Component Queues" on page 781. For general information about wildcards, see the *Junos OS Administration Library*.

NOTE: The interface applies wildcard configuration only if you do not add any specific configuration. If you add the specific interface configuration, then the interface deletes the applied wildcard configuration and applies the specified configuration.

For more information, see the following sections:

- Applying Custom Schedulers to Packet Forwarding Component Queues on page 780
- Examples: Scheduling Packet Forwarding Component Queues on page 781

Applying Custom Schedulers to Packet Forwarding Component Queues

Optionally, you can apply a custom scheduler to the chassis queues instead of configuring the chassis queues to automatically derive their scheduling configuration from the logical interfaces on the PIC.

To apply a custom chassis scheduler:

1. Specify the interface on which to apply the scheduler.

```
[edit]
user@host# edit class-of-service interfaces interface-name
```

   For example:

```
[edit]
user@host# edit class-of-service interfaces so-0/1/*
```

2. Specify the name of the custom scheduler you want to apply to the interface.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces so-0/1/0]
user@host# set scheduler-map-chassis map-name
```

When you apply a custom scheduler map to packet forwarding component queues, or when you modify the configuration of a custom scheduler map that is already applied to packet forwarding component queues, packets already in the chassis queues might be dropped. The amount of packet loss is not deterministic and depends on the offered traffic load at the time you apply or modify the custom scheduler map.

Examples: Scheduling Packet Forwarding Component Queues

- Example: Applying a Chassis Scheduler Map to a 2-Port IQ PIC on page 781
- Example: Configuring ATM CoS with a Normal Scheduler and a Chassis Scheduler on page 783
- Example: Configuring Two T3 Interfaces on a Channelized DS3 IQ PIC on page 785
- Example: Applying Normal Schedulers to Two T3 Interfaces on page 786
- Example: Applying a Chassis Scheduler to Two T3 Interfaces on page 788

Example: Applying a Chassis Scheduler Map to a 2-Port IQ PIC

This example applies a chassis scheduler map to interfaces **so-0/1/0** and **so-0/1/1**.

According to customary wildcard rules, the **so-0/1/0** configuration overrides the **so-0/1/*** configuration, implying that the chassis scheduler map **MAP1** is not applied to **so-0/1/0**. However, the wildcard rule is not obeyed in this case; the chassis scheduler map applies to both interfaces **so-0/1/0** and **so-0/1/1**.

To configure the chassis queues to derive their scheduling from the associated logical interfaces:

1. Specify the logical interfaces from which to derive the scheduling configuration.

   ```
   [edit class-of-service]
   user@host# set interfaces so-0/1/0 unit 0 classifiers inet-precedence default
   ```

2. Using a wildcard rule, specify that the scheduler configuration is derived from the logical interfaces on so-0/1*.

   ```
   [edit class-of-service]
   user@host# set interfaces so-0/1/* scheduler-map-chassis derived
   ```

3. Review the configuration.

   ```
   [edit]
   user@host# show
   ```

```
class-of-service {
  interfaces {
    so-0/1/0 {
      unit 0 {
        classifiers {
          inet-precedence default;
        }
      }
    }
    so-0/1/* {
      scheduler-map-chassis derived;
    }
  }
}
```

4. Save the configuration.

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

Not Recommended: Using a Wildcard for Gigabit Ethernet IQ Interfaces When Applying a Chassis Scheduler Map

On a Gigabit Ethernet IQ PIC, you can apply the chassis scheduler map at both the specific interface level and the wildcard level. We do not recommend this because the wildcard chassis scheduler map takes precedence, which might not be the desired effect. For example, if you want to apply the chassis scheduler map MAP1 to port 0 and MAP2 to port 1, we do not recommend the following:

```
[edit class-of-service]
user@host# set interfaces ge-0/1/0 scheduler-map-chassis MAP1
user@host# set interfaces ge-0/1/* scheduler-map-chassis MAP2
```

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

```
user@host# show
```

```
interfaces {
  ge-0/1/0 {
    scheduler-map-chassis MAP1;
  }
  ge-0/1/* {
    scheduler-map-chassis MAP2;
  }
}
```

Recommended: Identifying Gigabit Ethernet IQ Interfaces Individually When Applying a Chassis Scheduler Map

Instead, we recommend this configuration:

```
[edit class-of-service]
user@host# set interfaces ge-0/1/0 scheduler-map-chassis MAP1
user@host# set interfaces ge-0/1/1 scheduler-map-chassis MAP2
```

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


Copyright © 2019, Juniper Networks, Inc.

```
user@host# show
interfaces {
  ge-0/1/0 {
    scheduler-map-chassis MAP1;
  }
  ge-0/1/1 {
    scheduler-map-chassis MAP2;
  }
}
```

Example: Configuring ATM CoS with a Normal Scheduler and a Chassis Scheduler

For ATM2 IQ interfaces, the CoS configuration differs significantly from that of other interface types. For more information about ATM CoS, see "CoS on ATM Interfaces Overview" on page 841.

To configure scheduler mapping on ATM2 IQ interfaces:

1. Apply the chassis scheduler to the ATM interface.

   ```
   [edit]
   user@host# set class-of-service interfaces at-1/2/* scheduler-map-chassis derived
   ```

2. Configure ATM-specific physical interface properties.

   a. Specify the ATM interface to configure.

      ```
      [edit]
      user@host# edit interfaces at-1/2/* atm-options
      ```

   b. Configure the virtual path (VP).

      ```
      [edit interfaces at-1/2/* atm-options]
      user@host# set vpi 0
      ```

   c. Specify the CoS virtual circuit drop profiles for random early detection (RED). These parameters define the drop preferences during times of congestion.

      ```
      [edit interfaces at-1/2/0 atm-options]
      user@host# set linear-red-profiles red-profile-1 queue-depth 35k
      user@host# set linear-red-profiles red-profile-1 high-plp-threshold 75
      user@host# set linear-red-profiles red-profile-1 low-plp-threshold 25
      ```

   d. Define the CoS parameters for the scheduler map.

      ```
      [edit interfaces at-1/2/0 atm-options]
      user@host# set scheduler-maps map-1 vc-cos-mode strict
      user@host# set scheduler-maps map-1 forwarding-class best-effort priority low
      user@host# set scheduler-maps map-1 forwarding-class best-effort
        transmit-weight percent 25
      user@host# set scheduler-maps map-1 forwarding-class best-effort
        linear-red-profile red-profile-1
      ```

Copyright © 2019, Juniper Networks, Inc.

3. Configure the ATM logical interface.

   a. Specify the logical interface you want to configure.

   ```
   [edit interfaces at-1/2/0]
   user@host# edit unit 0
   ```

   b. Specify the virtual circuit identifier (VCI) and virtual path identifier (VPI) for the ATM logical interfaces.

   ```
   [edit interfaces at-1/2/0 unit 0]
   user@host# set vci 0.128
   ```

   c. Specify the traffic shaping profile parameters.

   ```
   [edit interfaces at-1/2/0 unit 0]
   user@host# set shaping vbr peak 20m sustained 10m burst 20
   ```

   d. Specify the scheduler map you want to associate with the ATM logical interface.

   ```
   [edit interfaces at-1/2/0 unit 0]
   user@host# set atm-scheduler-map map-1
   ```

   e. Configure the protocol, local address, and remote address.

   ```
   user@host# set family inet address 192.168.0.100/32 destination 192.168.0.101
   ```

4. Review the configuration.

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

```
user@host# show
```

```
interfaces {
  at-1/2/* {
    scheduler-map-chassis derived;
  }
}
```

```
[edit interfaces]
```

```
user@host# show
```

```
at-1/2/0 {
  atm-options {
    vpi 0;
    linear-red-profiles red-profile-1 {
      queue-depth 35000 high-plp-threshold 75 low-plp-threshold 25;
    }
    scheduler-maps map-1 {
      vc-cos-mode strict;
      forwarding-class best-effort {
        priority low;
        transmit-weight percent 25;
        linear-red-profile red-profile-1;
      }
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
  }
  unit 0 {
    vci 0.128;
    shaping {
      vbr peak 20m sustained 10m burst 20;
    }
    atm-scheduler-map map-1;
    family inet {
      address 192.168.0.100/32 {
        destination 192.168.0.101;
      }
    }
  }
}
```

5. Save the configuration.

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

Example: Configuring Two T3 Interfaces on a Channelized DS3 IQ PIC

To configure two T3 interfaces on a channelized DS3 IQ PIC:

1. Configure the first channelized DS3 IQ interface.
   a. Specify the name of the interface.

      ```
      [edit]
      user@host# edit interfaces ct3-3/0/0
      ```

   b. Configure the interface as unpartitioned, clear channel.

      ```
      [edit interfaces ct3-3/0/0]
      user@host# set no-partition interface-type t3
      ```

2. Configure the second channelized DS3 IQ interface.
   a. Specify the name of the interface.

      ```
      [edit]
      user@host# edit interfaces ct3-3/0/1
      ```

   b. Configure the interface as unpartitioned, clear channel.

      ```
      [edit interfaces ct3-3/0/1]
      user@host# set no-partition interface-type t3
      ```

3. Configure the first T3 channel.
   a. Specify the name of the T3 interface on the DS3 IQ PIC.

      ```
      [edit]
      user@host# edit t3-3/0/0 unit 0
      ```

Copyright © 2019, Juniper Networks, Inc.

b. Specify the protocol family and address of the interface.

```
[edit t3-3/0/0 unit 0]
user@host# set family inet address 10.0.100.1/30
```

4. Configure the second T3 channel.

a. Specify the name of the T3 interface on the DS3 IQ PIC.

```
[edit]
user@host# edit t3-3/0/1 unit 0
```

b. Specify the protocol family and address of the interface.

```
[edit t3-3/0/0 unit 0]
user@host# set family inet address 10.0.101.1/30
```

5. Review the configuration.

```
[edit interfaces]
```

```
user@host# show
ct3-3/0/0 {
  no-partition interface-type t3; # use entire port 0 as T3
}
ct3-3/0/1 {
  no-partition interface-type t3; # use entire port 1 as T3
}
t3-3/0/0 {
  unit 0 {
    family inet {
      address 10.0.100.1/30;
    }
  }
}
t3-3/0/1 {
  unit 0 {
    family inet {
      address 10.0.101.1/30;
    }
  }
}
```

6. Save the configuration.

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

Example: Applying Normal Schedulers to Two T3 Interfaces

Configure a scheduler for the aggregated traffic transmitted into both T3 interfaces.

1. Specify the names of the scheduler maps for each interface.

```
[edit]
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set class-of-service interfaces t3-3/0/0 scheduler-map sched-qct3-0
user@host# set class-of-service interfaces t3-3/0/1 scheduler-map sched-qct3-1
```

2. Specify the CoS parameters assigned to each forwarding class.

```
[edit]
user@host# set class-of-service scheduler-maps sched-qct3-0 forwarding-class
  best-effort scheduler be-qct3-0
user@host# set class-of-service scheduler-maps sched-qct3-0 forwarding-class
  expedited-forwarding scheduler ef-qct3-0
user@host# set class-of-service scheduler-maps sched-qct3-0 forwarding-class
  assured-forwarding scheduler as-qct3-0
user@host# set class-of-service scheduler-maps sched-qct3-0 forwarding-class
  network-control scheduler nc-qct3-0
user@host# set class-of-service scheduler-maps sched-qct3-1 forwarding-class
  best-effort scheduler be-qct3-1
user@host# set class-of-service scheduler-maps sched-qct3-1 forwarding-class
  expedited-forwarding scheduler ef-qct3-1
user@host# set class-of-service scheduler-maps sched-qct3-1 forwarding-class
  assured-forwarding scheduler as-qct3-1
user@host# set class-of-service scheduler-maps sched-qct3-1 forwarding-class
  network-control scheduler nc-qct3-1
user@host# set class-of-service scheduler-maps sched-chassis-to-q forwarding-class
  best-effort scheduler be-chassis
user@host# set class-of-service scheduler-maps sched-chassis-to-q forwarding-class
  expedited-forwarding scheduler ef-chassis
user@host# set class-of-service scheduler-maps sched-chassis-to-q forwarding-class
  assured-forwarding scheduler as-chassis
user@host# set class-of-service scheduler-maps sched-chassis-to-q forwarding-class
  network-control scheduler nc-chassis
```

3. Specify each scheduler name and the associated transmit rate..

```
user@host# set class-of-service schedulers be-qct3-0 transmit-rate percent 40
user@host# set class-of-service schedulers ef-qct3-0 transmit-rate percent 30
user@host# set class-of-service schedulers as-qct3-0 transmit-rate percent 20
user@host# set class-of-service schedulers nc-qct3-0 transmit-rate percent 10
```

4. Review the configuration.

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

```
user@host# show
```

```
interfaces {
  t3-3/0/0 {
    scheduler-map sched-qct3-0;
  }
  t3-3/0/1 {
    scheduler-map sched-qct3-1;
  }
```

Copyright © 2019, Juniper Networks, Inc.

```
}
scheduler-maps {
  sched-qct3-0 {
    forwarding-class best-effort scheduler be-qct3-0;
    forwarding-class expedited-forwarding scheduler ef-qct3-0;
    forwarding-class assured-forwarding scheduler as-qct3-0;
    forwarding-class network-control scheduler nc-qct3-0;
  }
  sched-qct3-1 {
    forwarding-class best-effort scheduler be-qct3-1;
    forwarding-class expedited-forwarding scheduler ef-qct3-1;
    forwarding-class assured-forwarding scheduler as-qct3-1;
    forwarding-class network-control scheduler nc-qct3-1;
  }
  sched-chassis-to-q {
    forwarding-class best-effort scheduler be-chassis;
    forwarding-class expedited-forwarding scheduler ef-chassis;
    forwarding-class assured-forwarding scheduler as-chassis;
    forwarding-class network-control scheduler nc-chassis;
  }
}
schedulers {
  be-qct3-0 {
    transmit-rate percent 40;
  }
  ef-qct3-0 {
    transmit-rate percent 30;
  }
  as-qct3-0 {
    transmit-rate percent 20;
  }
  nc-qct3-0 {
transmit-rate percent 10;
  }
}
```

5. Save the configuration.

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

Example: Applying a Chassis Scheduler to Two T3 Interfaces

Bind a scheduler to the aggregated traffic transmitted into the entire PIC. The chassis scheduler controls the traffic from the packet forwarding components feeding the interface **t3-3/0/***:

1. Using a wildcard rule, specify that the scheduler configuration is derived from the logical interfaces on t3-3/0/*:

```
user@host# set class-of-service interfaces t3-3/0/* scheduler-map-chassis derived
```

2. Review the configuration.

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


Copyright © 2019, Juniper Networks, Inc.

```
user@host# show

interfaces {
  t3-3/0/* {
    scheduler-map-chassis derived;
  }
}
```

3. Save the configuration.

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

Not Recommended: Using a Wildcard for Logical Interfaces When Applying a Scheduler

Do not apply a scheduler to a logical interface using a wildcard. For example, if you configure a logical interface (unit) with one parameter, and apply a scheduler map to the interface using a wildcard, the logical interface will not apply the scheduler. The following configuration will commit correctly but will not apply the scheduler map to interface **so-3/0/0.0**:

```
[edit]
user@host# set class-of-service interfaces so-3/0/* unit 0 scheduler-map
  MY_SCHED_MAP
user@host# set class-of-service interfaces so-3/0/0 unit 0 shaping-rate 100m
```

```
[edit class of service]
user@host# show
interfaces {
  so-3/0/* {
    unit 0 {
      scheduler-map MY_SCHED_MAP;
    }
  }
  so-3/0/0 {
    unit 0 {
      shaping-rate 100m;
    }
  }
}
```

Recommended: Identifying Logical Interfaces Individually When Applying a Scheduler

Always apply the scheduler to a logical interface without the wildcard:

```
[edit]
user@host# set class-of-service interfaces so-3/0/0 unit 0 scheduler-map
  MY_SCHED_MAP
```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set class-of-service interfaces so-3/0/0 unit 0 shaping-rate 100m
```

```
[edit class of service]
user@host# show
interfaces {
  so-3/0/0 {
    unit 0 {
      scheduler-map MY_SCHED_MAP;
      shaping-rate 100m;
    }
  }
}
```

NOTE: This same wildcard behavior applies to classifiers and rewrites as well as schedulers.

Related Documentation

- Configuring Scheduler Maps on page 258
- Applying Scheduler Maps to Physical Interfaces on page 259
- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300

Assigning Default Frame Relay Rewrite Rule to IQE PICs

On the Enhanced IQ (IQE) PICs with the Frame Relay encapsulation, you can rewrite the discard eligibility (DE) bit based on the loss priority of the Frame Relay traffic. A rewrite rule sets the DE bit to the class-of-service (CoS) value **0** or **1**, based on the assigned loss priority of low, medium-low, medium-high, or high for each outgoing frame.

The default Frame Relay rewrite rule contains the following settings:

```
loss-priority low code-point 0;
loss-priority medium-low code-point 0;
loss-priority medium-high code-point 1;
loss-priority high code-point 1;
```

The default rule sets the DE CoS value to **0** for each outgoing frame with the loss priority set to low or medium-low. The default rule sets the DE CoS value to **1** for each outgoing frame with the loss priority set to medium-high or high.

To assign the default Frame Relay rewrite rule to an interface:

1. Include the **frame-relay-de default** statement at the **[edit class-of-service interfaces interface *interface-name* unit *logical-unit-number* loss-priority-rewrites]** hierarchy level.

   For example:

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces so-1/0/0 unit 0 loss-priority-rewrites]
user@host# set frame-relay-de default;
```

2. Verify the configuration in operational mode.

```
user@host> show class-of-service loss-priority-rewrite
Loss-priority-rewrite: frame-relay-de-default, Code point type: frame-relay-de,
 Index: 38
  Loss priority      Code point
  low                 0
  high                1
  medium-low          0
  medium-high         1
```

Related Documentation

- *Frame Relay Overview*
- show class-of-service loss-priority-rewrite on page 1395

Defining Custom Frame Relay Rewrite Rule on IQE PICs

For Juniper Networks device interfaces with the Frame Relay encapsulation, you can rewrite the discard eligibility (DE) bit based on the loss priority of the Frame Relay traffic. A rewrite rule sets the DE bit to the class-of-service (CoS) value **0** or **1** based on the assigned loss priority of low, medium-low, medium-high, or high for each outgoing frame.

To define a Frame Relay DE bit rewrite rule:

1. Specify the rewrite rule for Frame Relay DE bit based on the loss priority at the **[edit class-of-service loss-priority-rewrites]** hierarchy level.

```
[edit class-of-service loss-priority-rewrites]
user@host# set frame-relay-de name loss-priority level code-point [ alias | bits ];
```

For example:

```
[edit class-of-service loss-priority-rewrites]
user@host# set frame-relay-de fr_rw loss-priority low code-point 0;
user@host# set frame-relay-de fr_rw loss-priority high code-point 0;
user@host# set frame-relay-de fr_rw loss-priority medium-low code-point 1;
user@host# set frame-relay-de fr_rw loss-priority medium-high code-point 1;
```

NOTE: The rewrite rule does not take effect until you apply it to a logical interface.

2. Apply a rule to a logical interface.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces interface-name unit logical-unit-number
  loss-priority-rewrites]
user@host# set frame-relay-de name;
```

For example:

```
[edit class-of-service interfaces so-1/0/0 unit 0 loss-priority-rewrites]
user@host# set frame-relay-de fr_rw;
```

3. Verify the configuration in operational mode.

```
user@host> show class-of-service loss-priority-rewrite
Loss-priority-rewrite: frame-relay-de-fr_rw, Code point type: frame-relay-de,
 Index: 38
  Loss priority      Code point
  low                 0
  high                0
  medium-low          1
  medium-high         1
```

Related Documentation

- frame-relay-de on page 1146
- show class-of-service loss-priority-rewrite on page 1395

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 21

Configuring Class of Service on Ethernet IQ2 and Enhanced IQ2 PICs

- CoS on Enhanced IQ2 PICs Overview on page 794
- CoS Features and Limitations on IQ2 and IQ2E PICs (M Series and T Series) on page 795
- Differences Between Gigabit Ethernet IQ and Gigabit Ethernet IQ2 PICs on page 796
- Shaping Granularity Values for Enhanced Queuing Hardware on page 798
- Ethernet IQ2 PIC RTT Delay Buffer Values on page 800
- Configuring BA Classifiers for Bridged Ethernet on page 801
- Setting the Number of Egress Queues on IQ2 and Enhanced IQ2 PICs on page 803
- Configuring the Number of Schedulers per Port for Ethernet IQ2 PICs on page 803
- Applying Scheduler Maps to Chassis-Level Queues on page 805
- Configuring a Policer Overhead on page 817
- CoS for L2TP Tunnels on Ethernet Interface Overview on page 818
- Configuring CoS for L2TP Tunnels on Ethernet Interfaces on page 819
- Configuring LNS CoS for Link Redundancy on page 820
- Example: Configuring L2TP LNS CoS Support for Link Redundancy on page 821
- Configuring Shaping on 10-Gigabit Ethernet IQ2 PICs on page 824
- Configuring Per-Unit Scheduling for GRE Tunnels Using IQ2 and IQ2E PICs on page 826
- Understanding Burst Size Configuration on IQ2 and IQ2E Interfaces on page 828
- Configuring Burst Size for Shapers on IQ2 and IQ2E Interfaces on page 829
- Configuring a CIR and a PIR on Ethernet IQ2 Interfaces on page 830
- Example: Configuring Shared Resources on Ethernet IQ2 Interfaces on page 832
- Configuring and Applying IEEE 802.1ad Classifiers on page 836
- Configuring Rate Limits to Protect Lower Queues on IQ2 and Enhanced IQ2 PICs on page 837
- Simple Filters Overview on page 839
- Configuring a Simple Filter on page 839

Copyright © 2019, Juniper Networks, Inc.

CoS on Enhanced IQ2 PICs Overview

Some PICs, such as the Gigabit Ethernet Intelligent Queuing 2 (IQ2) and Ethernet Enhanced IQ2 (IQ2E) PICs, have eight egress queues enabled by default on platforms that support eight queues.

The IQ2E PICs preserve all of the features of the IQ2 PICs, such as the default support for eight egress queues on platforms that support eight queues.

The IQ2E PICs add features such as the ability to perform hierarchical scheduling. You can mix IQ2 and IQ2E PICs on the same router.

The IQ2E PICs offer:

- Three levels of hierarchical CoS
- More granularity than a high priority queue
- 16,000 queues
- 2,000 schedulers with 8 queues
- 4,000 schedulers with 4 queues

The IQ2E PICs also offer automatic scheduler allocation across ports, so there is no need to reset the PIC when this changes. Random early detection (RED) keeps statistics on a per-drop-profile basis, improving the ability to perform network capacity planning.

When you include the **per-unit-scheduler** statement at the **[edit interfaces *interface-name*]** hierarchy level, each logical interface (unit) gets a dedicated scheduler (one scheduler is reserved for overflow). You can include the **per-session-scheduler** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level to shape Layer 2 Tunneling Protocol (L2TP) sessions. The behavior of these two-port scheduler modes is the same as in IQ2 PICs. However, IQ2E PICs use hierarchical schedulers and not shared schedulers; IQ2E PICs do not support the **shared-scheduler** statement at the **[edit interfaces *interface-name*]** hierarchy level.

For more information about configuring hierarchical schedulers, including examples, see "Configuring Hierarchical Schedulers for CoS" on page 342.

You can shape traffic at the physical interface (port), logical interface (unit), or interface set (set of units) levels. Shaping is not supported at the queue level. However, you can include the **transmit-rate** statement with the **rate-limit** option at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level to police the traffic passing through a queue (but only in the egress direction). See "Configuring Rate Limits to Protect Lower Queues on IQ2 and Enhanced IQ2 PICs" on page 837.

At the physical interface (port) level, you can configure only a shaping rate (PIR). At the logical interface (unit) and interface set levels, you can configure both a shaping rate and a guaranteed rate (CIR). Note that the guaranteed rates at any level must be consistent with the parent level's capacity. In other words, the sum of the guaranteed rates on the logical interface (units) should be less than the guaranteed rate on the

Copyright © 2019, Juniper Networks, Inc.

interface set, and the sum of the guaranteed rates on the logical interface (units) and interface sets should be less than the guaranteed rate on the physical interface (port).

You can control the rate of traffic that passes through the interface by configuring a policer overhead. When you configure a policer overhead, the configured policer overhead value is added to the length of the final Ethernet frame. This calculated length of the frame is used to determine the policer or the rate limit action. It does this because the policer overhead needs to be applied to policers just like shaping overhead is accounted for by shapers. The policer overhead is to be configured on the interface so that it is accounted for in the total packet length when policing traffic. See "Configuring a Policer Overhead" on page 817

The weighed RED (WRED) decision on the IQ2E PICs is done at the queue level. Once the accept or drop decision is made and the packet is queued, it is never dropped. Four drop profiles are associated with each queue: low, low-medium, medium-high, and high. WRED statistics are available for each loss priority (this feature is not supported on the IQ2 PICs). Also in contrast to the IQ2 PICs, the IQ2E PICs support WRED scaling profiles, allowing a single drop profile to be reused with a wide range of values. This practice increases the effective number of WRED drop profiles.

The IQ2E PICs provide four levels of strict priorities: strict-high, high, medium-high (medium-low) and low. In contrast to the IQ2 PICs, which support only one strict-high queue, the IQ2E PICs do not restrict the number of queues with a given priority. There is priority propagation among three levels: the logical interface, the logical interface set, and the physical port. These features are the same as those supported by Enhanced Queuing Dense Port Concentrators (DPCs) for Juniper Network MX Series 5G Universal Routing Platforms. For more information about configuring these features, see "Enhanced Queuing DPC CoS Properties" on page 908.

The IQ2E PIC's queues are serviced with modified deficit round-robin (MDRR), as with the Enhanced Queuing DPCs. Excess bandwidth (bandwidth available after all guaranteed rates have been satisfied) can be shared equally or in proportion to the guaranteed rates. For more information about excess bandwidth sharing, see "Configuring Excess Bandwidth Sharing" on page 916.

Related Documentation

- egress-policer-overhead on page 1097
- ingress-policer-overhead on page 1167

CoS Features and Limitations on IQ2 and IQ2E PICs (M Series and T Series)

This topic describes CoS scaling and performance parameters that apply to IQ2 and IQ2E PICs on M series and T series routers.

Classification

Behavior aggregate (BA) classification is done on the PIC. There are eight classifier tables of each type (ieee-802.1p, mpls-exp, inet-precedence, dscp, and dscp-ipv6) supported per PIC.

Copyright © 2019, Juniper Networks, Inc.

For each classifier type, one table is reserved for a default classifier. This table is used when no classifier is configured, or when the number of tables configured exceeds eight.

The following restrictions apply:

- You can only use BA classifiers for IPv4 DSCP bits for virtual private LAN service (VPLS).
- You cannot use BA classifiers for IPv4 DSCP bits for Layer 2 VPNs.
- You cannot use BA classifiers for IPv6 DSCP bits for VPLS.
- You cannot use BA classifiers for IPv6 DSCP bits for Layer 2 VPNs.

Rewrite Operations

802.1p or 802.1ad rewrite operations are done on the PIC. A total of eight rewrite markers of each type are supported on the PIC. For other rewrite operations, the numbers are the same as for any other M series FPCs. See "CoS Features and Limitations on M Series and T Series Routers" on page 553for details.

Differences Between Gigabit Ethernet IQ and Gigabit Ethernet IQ2 PICs

Because Gigabit Ethernet IQ PICs and Gigabit Ethernet IQ2 PICs use different architectures, they differ in the following ways:

- Gigabit Ethernet IQ2 PICs support a transmission rate within a queue, but do not support an exact rate within a queue. You can apply a weight to a queue, but you cannot put an upper limit on the queue transmission rate that is less than the logical interface can support. Consequently, including the **exact** option with the **transmit-rate (*rate* | percent *percent*)** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level is not supported for Gigabit Ethernet IQ2 interfaces.
- Gigabit Ethernet IQ2 PICs support only one queue in the scheduler map with **medium-high**, **high**, or **strict-high** priority. If more than one queue is configured with **medium-high**, **high**, or **strict-high** priority, the commit operation fails.
- To ensure that protocol control traffic (such as OSPF, BGP, and RIP) are not dropped at the oversubscribed ingress direction, the software puts control protocol packets into a separate control scheduler. There is one control scheduler per port. These control schedulers are implemented as strict-high priority, so they transmit traffic until they are empty.
- On Gigabit Ethernet IQ2 PICs, you can configure a single traffic-control profile to contain both a PIR (the **shaping-rate** statement) and a CIR (the **guaranteed-rate** statement). On Gigabit Ethernet IQ PICs, these statements are mutually exclusive.
- Gigabit Ethernet IQ2 PICs support only two fill levels in the RED drop profile. The recommended definition of the RED drop profile is as follows:

```
class-of-service {
  drop-profiles {
    drop-iq2-example1 {
      fill-level 20 drop-probability 0;
      fill-level 100 drop-probability 80;
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
  }
}
```

This configuration defines a drop profile with a linear drop probability curve when the fill level is between 20 and 100 percent, and a maximum drop probability of 80 percent.

You can configure more than two fill levels in a drop profile, but the software only uses the points (**min_fill_level**, **0**) and (**max_fill_level**, **max_probability**) and ignores other fill levels. The drop probability at the minimum fill level is set to 0 percent even if you configure a non-zero drop probability value at the minimum fill level. The following example shows a sample configuration and the software implementation:

Configuration

```
class-of-service {
  drop-profiles {
    drop-iq2-example2 {
      fill-level 30 drop-probability 10;
      fill-level 40 drop-probability 20;
      fill-level 100 drop-probability 80;
    }
  }
}
```

Implementation

```
class-of-service {
  drop-profiles {
    drop-iq2-example2-implementation {
      fill-level 30 drop-probability 0;
      fill-level 100 drop-probability 80;
    }
  }
}
```

If you configure more than two fill levels, a system log message warns you that the software supports only two fill levels and displays the drop profile that is implemented.

Though the **interpolate** statement is supported in the definition of a RED drop profile, we do not recommend using it. The following example shows a sample configuration and the software implementation:

Configuration

```
class-of-service {
  drop-profiles {
    drop-iq2-example3 {
      interpolate {
        fill-level [ 30 50 80 ];
        drop-probability [ 10 20 40 ];
      }
    }
  }
}
```

Copyright © 2019, Juniper Networks, Inc.

When you use the **interpolate** statement and the maximum fill level is not 100 percent, the software adds the point (100, 100). Therefore, the drop-iq2-example3 drop profile is implemented as:

Implementation

```
class-of-service {
  drop-profiles {
    drop-iq2-example3-implementation {
      fill-level 2 drop-probability 0;
      fill-level 100 drop-probability 100;
    }
  }
}
```

The implemented minimum fill level is not 30 percent as configured, but 2 percent because of the 64-point interpolation.

Shaping Granularity Values for Enhanced Queuing Hardware

Due to the limits placed on shaping thresholds used in the hierarchy, there is a granularity associated with the Enhanced IQ2 (IQ2E) PIC and the Enhanced Queuing (EQ) DPC. For these hardware models, the shaper accuracies differ at various levels of the hierarchy, with shapers at the logical interface level (Level 3) being more accurate than shapers at the interface set level (Level 2) or the port level (Level 1). Table 109 on page 798 shows the accuracy of the logical interface shaper at various rates for Ethernet ports operating at 1 Gbps.

Table 109: Shaper Accuracy of 1-Gbps Ethernet at the Logical Interface Level

Range of Logical Interface Shaper	Step Granularity
Up to 4.096 Mbps	16 Kbps
4.096 to 8.192 Mbps	32 Kbps
8.192 to 16.384 Mbps	64 Kbps
16.384 to 32.768 Mbps	128 Kbps
32.768 to 65.535 Mbps	256 Kbps
65.535 to 131.072 Mbps	512 Kbps
131.072 to 262.144 Mbps	1024 Kbps
262.144 to 1 Gbps	4096 Kbps

Table 110 on page 799 shows the accuracy of the logical interface shaper at various rates for Ethernet ports operating at 10 Gbps.

Copyright © 2019, Juniper Networks, Inc.

Table 110: Shaper Accuracy of 10-Gbps Ethernet at the Logical Interface Level

Range of Logical Interface Shaper	Step Granularity
Up to 10.24 Mbps	40 Kbps
10.24 to 20.48 Mbps	80 Kbps
10.48 to 40.96 Mbps	160 Kbps
40.96 to 81.92 Mbps	320 Kbps
81.92 to 163.84 Mbps	640 Kbps
163.84 to 327.68 Mbps	1280 Kbps
327.68 to 655.36 Mbps	2560 Kbps
655.36 to 2611.2 Mbps	10240 Kbps
2611.2 to 5222.4 Mbps	20480 Kbps
5222.4 to 10 Gbps	40960 Kbps

Table 111 on page 799 shows the accuracy of the interface set shaper at various rates for Ethernet ports operating at 1 Gbps.

Table 111: Shaper Accuracy of 1-Gbps Ethernet at the Interface Set Level

Range of Interface Set Shaper	Step Granularity
Up to 20.48 Mbps	80 Kbps
20.48 Mbps to 81.92 Mbps	320 Kbps
81.92 Mbps to 327.68 Mbps	1.28 Mbps
327.68 Mbps to 1 Gbps	20.48 Mbps

Table 112 on page 799 shows the accuracy of the interface set shaper at various rates for Ethernet ports operating at 10 Gbps.

Table 112: Shaper Accuracy of 10-Gbps Ethernet at the Interface Set Level

Range of Interface Set Shaper	Step Granularity
Up to 128 Mbps	500 Kbps
128 Mbps to 512 Mbps	2 Mbps
512 Mbps to 2.048 Gbps	8 Mbps

Copyright © 2019, Juniper Networks, Inc.

Table 112: Shaper Accuracy of 10-Gbps Ethernet at the Interface Set Level (continued)

Range of Interface Set Shaper	Step Granularity
2.048 Gbps to 10 Gbps	128 Mbps

Table 113 on page 800 shows the accuracy of the physical port shaper at various rates for Ethernet ports operating at 1 Gbps.

Table 113: Shaper Accuracy of 1-Gbps Ethernet at the Physical Port Level

Range of Physical Port Shaper	Step Granularity
Up to 64 Mbps	250 Kbps
64 Mbps to 256 Mbps	1 Mbps
256 Mbps to 1 Gbps	4 Mbps

Table 114 on page 800 shows the accuracy of the physical port shaper at various rates for Ethernet ports operating at 10 Gbps.

Table 114: Shaper Accuracy of 10-Gbps Ethernet at the Physical Port Level

Range of Physical Port Shaper	Step Granularity
Up to 640 Mbps	2.5 Mbps
640 Mbps to 2.56 Gbps	10 Mbps
2.56 Gbps to 10 Gbps	40 Mbps

Ethernet IQ2 PIC RTT Delay Buffer Values

The following table shows the round-trip time (RTT) delay buffer values for IQ2 PICs, which are nonstandard and vary by PIC type and direction. The values are rounded up slightly to account for oversubscription.

Table 115: RTT Delay Buffers for IQ2 PICs

IQ2 PIC Type	Ingress Buffer (ms)	Egress Buffer (ms)
4–port Gigabit Ethernet (Type 1)	200	300
8–port Gigabit Ethernet (Type 2)	175	200
8–port Gigabit Ethernet (Type 3)	35	225
1–port 10–Gigabit Ethernet (Type 3)	25	190

Copyright © 2019, Juniper Networks, Inc.

Configuring BA Classifiers for Bridged Ethernet

On M120 and M320 routers equipped with IQ2 PICs, you can configure BA classification based on the IEEE 802.1 bits for bridged Ethernet over Asynchronous Transfer Mode (ATM), Point-to-Point Protocol (PPP), and frame relay for VPLS applications. The BA classification is applied to the first (outer) tag when tagged frames are received. Untagged frames are bypassed and a value of 000 for the classification IEEE 802.1p bits is assumed. There is no support for circuit cross-connect (CCC), and only port-mode VPLS is supported (in port-mode VPLS, only VLANs on a single physical port are included in the VPLS instance). There is no support for multilink PPP bonding with VPLS. For bridging over frame relay, only frames that do not preserve the frame check sequence (FCS) field are supported. Frames that preserve the FCS field are silently discarded.

The bridging over PPP function is restricted:

- There is no support for “tinygram” compression and expansion.
- Frames received with preserved FCS bits are silently discarded.
- Bridge control frames are also classified based on header bit values.
- Both tagged and untagged frames are classified and forwarded. The peer must discard frame types that are not supported.

The following example applies an IEEE 802.1p classifier named **ppp-ether-vpls-classifier** to interface (**so-1/2/3**) with Ethernet VPLS over PPP encapsulation.

NOTE: The interface and CoS configuration must be consistent to support the feature. You must also configure the classifier and other CoS parameters such as forwarding classes.

1. Apply the CoS behavior aggregate classifier to a logical interface.

```
[edit]
user@host# set class-of-service interfaces so-1/2/3 unit 0 classifiers ieee-802.1
  ppp-ether-vpls-classifier
```

2. Configure the encapsulation and protocol family for the interface.

```
[edit]
user@host# set interfaces so-1/2/3 encapsulation ethernet-vpls-ppp unit 0 family
  vpls
```

3. Verify the configuration.

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

```
user@host# show
```

```
interfaces {
  so-1/2/3 {
   unit 0 {
    classifiers {
     ieee-802.1 ppp-ether-vpls-classifier;
    }
   }
  }
}
```

```
[edit interfaces]
```

```
user@host# show
```

```
s0-1/2/3 {
  encapsulation ether-vpls-over-ppp;
  unit 0 {
   family vpls;
  }
}
```

4. Save the configuration.

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

On routers with IQ2 or IQ2E PICs, you can perform BA classification based on the value of the inner VLAN tag in an Ethernet frame.

1. To configure BA classification based on the inner VLAN tag value, speciy the **inner** option at the [**edit class-of-service interfaces** *interface-name* **unit** *logical-unit-number* **classifiers ieee-802.1** *classifier-name* **vlan-tag**] hierarchy level. You must also configure the inner VLAN tag for the logical interface with the ***inner*** option at the [**edit interfaces** *interface-name* **unit** *logical-interface-name* **vlan-tag]** hierarchy level.

```
[edit]
user@host# set class-of-service interfaces ge-2/2/2 unit 0 classifiers ieee-802.1
  inner-vlan-tag-ba-classifier
user@host# set class-of-service interfaces ge-2/2/2 unit 0 classifiers ieee-802.1
  vlan-tag inner
```

2. Verify the configuration.

```
[edit]
```

```
user@host# show
```

```
class-of-service {
 interfaces {
   ge-2/2/2 unit 0
    classifiers ieee-802.1 inner-vlan-tag-ba-classifier {
     vlan-tag inner;
    }
```


Copyright © 2019, Juniper Networks, Inc.

```
  }
}
```

3. Save the configuration.

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

Related Documentation

- CoS on Enhanced IQ2 PICs Overview on page 794
- Configuring Behavior Aggregate Classifiers on page 51
- Default IEEE 802.1p Classifier on page 43

Setting the Number of Egress Queues on IQ2 and Enhanced IQ2 PICs

Gigabit Ethernet IQ2 4-port and 8-port Type 2 PICs are oversubscribed, which means the amount of traffic coming to the PIC can be more than the maximum bandwidth from the PIC to the Flexible PIC Concentrator (FPC).

By default, PICs on M320, MX Series, and T Series routers support a maximum of four egress queues per interface. Some PICs, such as the IQ2 and IQ2E PICs, have eight egress queues enabled by default on platforms that support eight queues. You configure the number of egress queues as four or eight by including the **max-queues-per-interface** statement at the **[edit chassis fpc *slot-number* pic *pic-slot-number*]** hierarchy level:

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

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

For more information about configuring egress queues, see "Configuring Up to 16 Custom Forwarding Classes" on page 214.

Configuring the Number of Schedulers per Port for Ethernet IQ2 PICs

You can oversubscribe the Ethernet IQ2 family of PICs. Because of the bursty nature of Ethernet use, traffic received by the PIC can be several orders of magnitude greater than the maximum bandwidth leaving the PIC and entering the router. Several configuration statements apply only to Ethernet IQ2 PICs and allow the PIC to intelligently handle the oversubscribed traffic.

NOTE: **The total of the input guaranteed rates for oversubscribed IQ2 PICs is limited to the FPC or PIC bandwidth.**

By default, each Ethernet IQ2 PIC is allocated a fixed number of the 1024 available schedulers for each port during PIC initialization. For example, the 8-port Gigabit Ethernet IQ2 PIC is allocated 128 schedulers for each port. This number cannot be changed after

Copyright © 2019, Juniper Networks, Inc.

the PIC is operational and can limit the utilization of shapers among the ports. Each of the 1024 schedulers is mapped at the logical interface (unit) level, and each scheduler map can support up to eight forwarding classes.

Schedulers are allocated in multiples of four. Three schedulers are reserved on each port. One is for control traffic, one is for port-level shaping, and the last is for unshaped logical interface traffic. These are allocated internally and automatically. The fourth scheduler is added when VLANs are configured.

When you configure schedulers for a port on an Ethernet IQ2 PIC:

- The three reserved schedulers are added to the configured value, which yields four schedulers per port.
- The configured value is adjusted upward to the nearest multiple of 4 (schedulers are allocated in multiples of 4).
- After all configured schedulers are allocated, any remaining unallocated schedulers are partitioned equally across the other ports.
- Any remaining schedulers that cannot be allocated meaningfully across the ports are allocated to the last port.

If the configured scheduler number is changed, the Ethernet IQ2 PIC is restarted when the configuration is committed.

CAUTION: If you deactivate and reactivate a port configured with a non-default number of schedulers, then the entire Ethernet IQ2 PIC restarts.

You can configure between 1 and 1024 schedulers on a port.

The following example allocates 100 schedulers to port 1 on an 8-port Gigabit Ethernet IQ2 PIC. The example shows the final scheduler allocation numbers for each port on the PIC. By default, each port would have been allocated 1024 / 8 = 128 schedulers.To configure the number of schedulers assigned to a port on an Ethernet IQ2 PIC:

- Specify the **schedulers** statement for the Ethernet IQ2 PIC interface at the **[edit interfaces ge-*fpc/pic/port*]** hierarchy level.

```
[edit interfaces ge-1/2/1]
user@host# set schedulers 100
```

This configuration results in the port and scheduler configuration shown in Table 116 on page 804.

Table 116: Scheduler Allocation for an Ethernet IQ2 PIC

Ethernet IQ2 PIC Port	Number of Allocated Schedulers
0	128

Copyright © 2019, Juniper Networks, Inc.

Table 116: Scheduler Allocation for an Ethernet IQ2 PIC (continued)

Ethernet IQ2 PIC Port	Number of Allocated Schedulers
1	104 (100 configured, plus 3 reserved, rounded up to multiple of 4: 100 + 3 +1= 104)
2	128
3	128
4	128
5	128
6	128
7	152 (128 plus the 24 remaining that cannot be meaningfully allocated to other ports)

Related Documentation

- How Schedulers Define Output Queue Properties on page 254

Applying Scheduler Maps to Chassis-Level Queues

On Intelligent Queuing (IQ) and Intelligent Queuing 2 (IQ2) interfaces, as well as on the 10x10GE MIC with SFP+, the traffic that is fed from the packet forwarding components into the PIC uses low packet loss priority (PLP) by default and is distributed evenly across the four chassis queues (not PIC queues), regardless of the scheduling configuration for each logical interface. This default behavior can cause traffic congestion.

The default chassis scheduler allocates resources for queue 0 through queue 3, with 25 percent of the bandwidth allocated to each queue. When you configure the chassis to use more than four queues, you must configure and apply a custom chassis scheduler to override the default chassis scheduler.

To apply a custom chassis scheduler:

1. Specify the interface on which to apply the scheduler.

```
[edit]
user@host# edit class-of-service interfaces interface-name
```

   For example:

```
[edit]
user@host# edit class-of-service interfaces so-0/1/*
```

2. Specify the name of the custom scheduler you want to apply to the interface.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces so-0/1/0]
user@host# set scheduler-map-chassis map-name
```

To control the aggregated traffic transmitted from the chassis queues into the PIC, you can configure the chassis queues to derive their scheduling configuration from the associated logical interface's.

To configure the chassis queues to derive their scheduling from the associated logical interfaces:

1. Specify the logical interfaces from which to derive the scheduling configuration.

   ```
   [edit]
   user@host# edit class-of-service interfaces interface-name
   ```

   For example:

   ```
   [edit]
   user@host# edit class-of-service interfaces so-0/1/*
   ```

2. Specify that the scheduler configuration is derived from the specified logical interfaces.

   ```
   [edit class-of-service interfaces so-0/1/0]
   user@host# set scheduler-map-chassis derived
   ```

CAUTION: If you specify the **scheduler-map-chassis derived** statement in the configuration, packet loss might occur when you subsequently add or remove logical interfaces at the **[edit interfaces *interface-name*]** hierarchy level.

When fragmentation occurs on the egress interface, the first set of packet counters displayed in the output of the **show interfaces queue** command show the post-fragmentation values. The second set of packet counters (under the **Packet Forwarding Engine Chassis Queues** field) show the pre-fragmentation values. For more information about the **show interfaces queue** command, see the CLI Explorer.

You can specify both the **scheduler-map** and the **scheduler-map-chassis derived** statements in the same interface configuration. The **scheduler-map** statement controls the scheduler inside the PIC, while the **scheduler-map-chassis derived** statement controls the aggregated traffic transmitted into the entire PIC.

NOTE: For the Gigabit Ethernet IQ PIC, you must specify both the **scheduler-map** and the **scheduler-map-chassis derived** statements in the interface configuration.

Copyright © 2019, Juniper Networks, Inc.

Generally, when you specify the **scheduler-map-chassis** statement in the configuration, you must use an interface wildcard for the interface name, as in ***type-fpc/pic/****. The wildcard must use this format—for example. **so-1/2/***, which means all interfaces on FPC slot 1, PIC slot 2. There is one exception—you can apply the chassis scheduler map to a specific interface on the Gigabit Ethernet IQ PIC only.

According to Junos OS wildcard rules, specific interface configurations override wildcard configurations. For chassis scheduler map configurations, this rule does not apply; instead, specific interface CoS configurations are added to the chassis scheduler map configuration. For more information about how wildcards work with chassis scheduler maps, see "Examples: Scheduling Packet Forwarding Component Queues" on page 781. For general information about wildcards, see the *Junos OS Administration Library*.

NOTE: **The interface applies wildcard configuration only if you do not add any specific configuration. If you add the specific interface configuration, then the interface deletes the applied wildcard configuration and applies the specified configuration.**

For more information, see the following sections:

- Applying Custom Schedulers to Packet Forwarding Component Queues on page 807
- Examples: Scheduling Packet Forwarding Component Queues on page 808

Applying Custom Schedulers to Packet Forwarding Component Queues

Optionally, you can apply a custom scheduler to the chassis queues instead of configuring the chassis queues to automatically derive their scheduling configuration from the logical interfaces on the PIC.

To apply a custom chassis scheduler:

1. Specify the interface on which to apply the scheduler.

```
[edit]
user@host# edit class-of-service interfaces interface-name
```

   For example:

```
[edit]
user@host# edit class-of-service interfaces so-0/1/*
```

2. Specify the name of the custom scheduler you want to apply to the interface.

```
[edit class-of-service interfaces so-0/1/0]
user@host# set scheduler-map-chassis map-name
```

When you apply a custom scheduler map to packet forwarding component queues, or when you modify the configuration of a custom scheduler map that is already applied to packet forwarding component queues, packets already in the chassis queues might

Copyright © 2019, Juniper Networks, Inc.

be dropped. The amount of packet loss is not deterministic and depends on the offered traffic load at the time you apply or modify the custom scheduler map.

Examples: Scheduling Packet Forwarding Component Queues

- Example: Applying a Chassis Scheduler Map to a 2-Port IQ PIC on page 808
- Example: Configuring ATM CoS with a Normal Scheduler and a Chassis Scheduler on page 810
- Example: Configuring Two T3 Interfaces on a Channelized DS3 IQ PIC on page 812
- Example: Applying Normal Schedulers to Two T3 Interfaces on page 813
- Example: Applying a Chassis Scheduler to Two T3 Interfaces on page 815

Example: Applying a Chassis Scheduler Map to a 2-Port IQ PIC

This example applies a chassis scheduler map to interfaces **so-0/1/0** and **so-0/1/1**.

According to customary wildcard rules, the **so-0/1/0** configuration overrides the **so-0/1/*** configuration, implying that the chassis scheduler map **MAP1** is not applied to **so-0/1/0**. However, the wildcard rule is not obeyed in this case; the chassis scheduler map applies to both interfaces **so-0/1/0** and **so-0/1/1**.

To configure the chassis queues to derive their scheduling from the associated logical interfaces:

1. Specify the logical interfaces from which to derive the scheduling configuration.

```
[edit class-of-service]
user@host# set interfaces so-0/1/0 unit 0 classifiers inet-precedence default
```

2. Using a wildcard rule, specify that the scheduler configuration is derived from the logical interfaces on so-0/1*.

```
[edit class-of-service]
user@host# set interfaces so-0/1/* scheduler-map-chassis derived
```

3. Review the configuration.

```
[edit]
```

```
user@host# show
```

```
class-of-service {
  interfaces {
    so-0/1/0 {
      unit 0 {
        classifiers {
          inet-precedence default;
        }
      }
    }
    so-0/1/* {
```


Copyright © 2019, Juniper Networks, Inc.

```
      scheduler-map-chassis derived;
    }
  }
}
```

4. Save the configuration.

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

Not Recommended: Using a Wildcard for Gigabit Ethernet IQ Interfaces When Applying a Chassis Scheduler Map

On a Gigabit Ethernet IQ PIC, you can apply the chassis scheduler map at both the specific interface level and the wildcard level. We do not recommend this because the wildcard chassis scheduler map takes precedence, which might not be the desired effect. For example, if you want to apply the chassis scheduler map MAP1 to port 0 and MAP2 to port 1, we do not recommend the following:

```
[edit class-of-service]
user@host# set interfaces ge-0/1/0 scheduler-map-chassis MAP1
user@host# set interfaces ge-0/1/* scheduler-map-chassis MAP2
```

```
[edit class-of-service]
user@host# show
interfaces {
  ge-0/1/0 {
    scheduler-map-chassis MAP1;
  }
  ge-0/1/* {
    scheduler-map-chassis MAP2;
  }
}
```

Recommended: Identifying Gigabit Ethernet IQ Interfaces Individually When Applying a Chassis Scheduler Map

Instead, we recommend this configuration:

```
[edit class-of-service]
user@host# set interfaces ge-0/1/0 scheduler-map-chassis MAP1
user@host# set interfaces ge-0/1/1 scheduler-map-chassis MAP2
```

```
[edit class-of-service]
user@host# show
interfaces {
  ge-0/1/0 {
    scheduler-map-chassis MAP1;
  }
  ge-0/1/1 {
    scheduler-map-chassis MAP2;
  }
}
```

Copyright © 2019, Juniper Networks, Inc.

Example: Configuring ATM CoS with a Normal Scheduler and a Chassis Scheduler

For ATM2 IQ interfaces, the CoS configuration differs significantly from that of other interface types. For more information about ATM CoS, see "CoS on ATM Interfaces Overview" on page 841.

To configure scheduler mapping on ATM2 IQ interfaces:

1. Apply the chassis scheduler to the ATM interface.

```
[edit]
user@host# set class-of-service interfaces at-1/2/* scheduler-map-chassis derived
```

2. Configure ATM-specific physical interface properties.

   a. Specify the ATM interface to configure.

```
[edit]
user@host# edit interfaces at-1/2/* atm-options
```

   b. Configure the virtual path (VP).

```
[edit interfaces at-1/2/* atm-options]
user@host# set vpi 0
```

   c. Specify the CoS virtual circuit drop profiles for random early detection (RED). These parameters define the drop preferences during times of congestion.

```
[edit interfaces at-1/2/0 atm-options]
user@host# set linear-red-profiles red-profile-1 queue-depth 35k
user@host# set linear-red-profiles red-profile-1 high-plp-threshold 75
user@host# set linear-red-profiles red-profile-1 low-plp-threshold 25
```

   d. Define the CoS parameters for the scheduler map.

```
[edit interfaces at-1/2/0 atm-options]
user@host# set scheduler-maps map-1 vc-cos-mode strict
user@host# set scheduler-maps map-1 forwarding-class best-effort priority low
user@host# set scheduler-maps map-1 forwarding-class best-effort
  transmit-weight percent 25
user@host# set scheduler-maps map-1 forwarding-class best-effort
  linear-red-profile red-profile-1
```

3. Configure the ATM logical interface.

   a. Specify the logical interface you want to configure.

```
[edit interfaces at-1/2/0]
user@host# edit unit 0
```

   b. Specify the virtual circuit identifier (VCI) and virtual path identifier (VPI) for the ATM logical interfaces.

Copyright © 2019, Juniper Networks, Inc.

```
[edit interfaces at-1/2/0 unit 0]
user@host# set vci 0.128
```

c. Specify the traffic shaping profile parameters.

```
[edit interfaces at-1/2/0 unit 0]
user@host# set shaping vbr peak 20m sustained 10m burst 20
```

d. Specify the scheduler map you want to associate with the ATM logical interface.

```
[edit interfaces at-1/2/0 unit 0]
user@host# set atm-scheduler-map map-1
```

e. Configure the protocol, local address, and remote address.

```
user@host# set family inet address 192.168.0.100/32 destination 192.168.0.101
```

4. Review the configuration.

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

```
user@host# show
```

```
interfaces {
  at-1/2/* {
    scheduler-map-chassis derived;
  }
}
```

```
[edit interfaces]
```

```
user@host# show
```

```
at-1/2/0 {
  atm-options {
    vpi 0;
    linear-red-profiles red-profile-1 {
      queue-depth 35000 high-plp-threshold 75 low-plp-threshold 25;
    }
    scheduler-maps map-1 {
      vc-cos-mode strict;
      forwarding-class best-effort {
        priority low;
        transmit-weight percent 25;
        linear-red-profile red-profile-1;
      }
    }
  }
  unit 0 {
    vci 0.128;
    shaping {
      vbr peak 20m sustained 10m burst 20;
    }
    atm-scheduler-map map-1;
    family inet {
      address 192.168.0.100/32 {
        destination 192.168.0.101;
```

Copyright © 2019, Juniper Networks, Inc.

```
      }
    }
  }
}
```

5. Save the configuration.

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

Example: Configuring Two T3 Interfaces on a Channelized DS3 IQ PIC

To configure two T3 interfaces on a channelized DS3 IQ PIC:

1. Configure the first channelized DS3 IQ interface.
   a. Specify the name of the interface.

      ```
      [edit]
      user@host# edit interfaces ct3-3/0/0
      ```

   b. Configure the interface as unpartitioned, clear channel.

      ```
      [edit interfaces ct3-3/0/0]
      user@host# set no-partition interface-type t3
      ```

2. Configure the second channelized DS3 IQ interface.
   a. Specify the name of the interface.

      ```
      [edit]
      user@host# edit interfaces ct3-3/0/1
      ```

   b. Configure the interface as unpartitioned, clear channel.

      ```
      [edit interfaces ct3-3/0/1]
      user@host# set no-partition interface-type t3
      ```

3. Configure the first T3 channel.
   a. Specify the name of the T3 interface on the DS3 IQ PIC.

      ```
      [edit]
      user@host# edit t3-3/0/0 unit 0
      ```

   b. Specify the protocol family and address of the interface.

      ```
      [edit t3-3/0/0 unit 0]
      user@host# set family inet address 10.0.100.1/30
      ```

4. Configure the second T3 channel.
   a. Specify the name of the T3 interface on the DS3 IQ PIC.

Copyright © 2019, Juniper Networks, Inc.

```
[edit]
user@host# edit t3-3/0/1 unit 0
```

b. Specify the protocol family and address of the interface.

```
[edit t3-3/0/0 unit 0]
user@host# set family inet address 10.0.101.1/30
```

5. Review the configuration.

```
[edit interfaces]
```

```
user@host# show
```

```
ct3-3/0/0 {
  no-partition interface-type t3; # use entire port 0 as T3
}
ct3-3/0/1 {
  no-partition interface-type t3; # use entire port 1 as T3
}
t3-3/0/0 {
  unit 0 {
    family inet {
      address 10.0.100.1/30;
    }
  }
}
t3-3/0/1 {
  unit 0 {
    family inet {
      address 10.0.101.1/30;
    }
  }
}
```

6. Save the configuration.

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

Example: Applying Normal Schedulers to Two T3 Interfaces

Configure a scheduler for the aggregated traffic transmitted into both T3 interfaces.

1. Specify the names of the scheduler maps for each interface.

```
[edit]
user@host# set class-of-service interfaces t3-3/0/0 scheduler-map sched-qct3-0
user@host# set class-of-service interfaces t3-3/0/1 scheduler-map sched-qct3-1
```

2. Specify the CoS parameters assigned to each forwarding class.

```
[edit]
```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set class-of-service scheduler-maps sched-qct3-0 forwarding-class
  best-effort scheduler be-qct3-0
user@host# set class-of-service scheduler-maps sched-qct3-0 forwarding-class
  expedited-forwarding scheduler ef-qct3-0
user@host# set class-of-service scheduler-maps sched-qct3-0 forwarding-class
  assured-forwarding scheduler as-qct3-0
user@host# set class-of-service scheduler-maps sched-qct3-0 forwarding-class
  network-control scheduler nc-qct3-0
user@host# set class-of-service scheduler-maps sched-qct3-1 forwarding-class
  best-effort scheduler be-qct3-1
user@host# set class-of-service scheduler-maps sched-qct3-1 forwarding-class
  expedited-forwarding scheduler ef-qct3-1
user@host# set class-of-service scheduler-maps sched-qct3-1 forwarding-class
  assured-forwarding scheduler as-qct3-1
user@host# set class-of-service scheduler-maps sched-qct3-1 forwarding-class
  network-control scheduler nc-qct3-1
user@host# set class-of-service scheduler-maps sched-chassis-to-q forwarding-class
  best-effort scheduler be-chassis
user@host# set class-of-service scheduler-maps sched-chassis-to-q forwarding-class
  expedited-forwarding scheduler ef-chassis
user@host# set class-of-service scheduler-maps sched-chassis-to-q forwarding-class
  assured-forwarding scheduler as-chassis
user@host# set class-of-service scheduler-maps sched-chassis-to-q forwarding-class
  network-control scheduler nc-chassis
```

3. Specify each scheduler name and the associated transmit rate..

```
user@host# set class-of-service schedulers be-qct3-0 transmit-rate percent 40
user@host# set class-of-service schedulers ef-qct3-0 transmit-rate percent 30
user@host# set class-of-service schedulers as-qct3-0 transmit-rate percent 20
user@host# set class-of-service schedulers nc-qct3-0 transmit-rate percent 10
```

4. Review the configuration.

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

```
user@host# show
```

```
interfaces {
  t3-3/0/0 {
    scheduler-map sched-qct3-0;
  }
  t3-3/0/1 {
    scheduler-map sched-qct3-1;
  }
}
scheduler-maps {
  sched-qct3-0 {
    forwarding-class best-effort scheduler be-qct3-0;
    forwarding-class expedited-forwarding scheduler ef-qct3-0;
    forwarding-class assured-forwarding scheduler as-qct3-0;
    forwarding-class network-control scheduler nc-qct3-0;
  }
```


Copyright © 2019, Juniper Networks, Inc.

```
  sched-qct3-1 {
    forwarding-class best-effort scheduler be-qct3-1;
    forwarding-class expedited-forwarding scheduler ef-qct3-1;
    forwarding-class assured-forwarding scheduler as-qct3-1;
    forwarding-class network-control scheduler nc-qct3-1;
  }
  sched-chassis-to-q {
    forwarding-class best-effort scheduler be-chassis;
    forwarding-class expedited-forwarding scheduler ef-chassis;
    forwarding-class assured-forwarding scheduler as-chassis;
    forwarding-class network-control scheduler nc-chassis;
  }
}
schedulers {
  be-qct3-0 {
    transmit-rate percent 40;
  }
  ef-qct3-0 {
    transmit-rate percent 30;
  }
  as-qct3-0 {
    transmit-rate percent 20;
  }
  nc-qct3-0 {
transmit-rate percent 10;
  }
}
```

5. Save the configuration.

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

Example: Applying a Chassis Scheduler to Two T3 Interfaces

Bind a scheduler to the aggregated traffic transmitted into the entire PIC. The chassis scheduler controls the traffic from the packet forwarding components feeding the interface **t3-3/0/***:

1. Using a wildcard rule, specify that the scheduler configuration is derived from the logical interfaces on t3-3/0/*.

```
user@host# set class-of-service interfaces t3-3/0/* scheduler-map-chassis derived
```

2. Review the configuration.

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

```
user@host# show
```

```

interfaces {
  t3-3/0/* {
    scheduler-map-chassis derived;
```

```
  }
}
```

3. Save the configuration.

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

Not Recommended: Using a Wildcard for Logical Interfaces When Applying a Scheduler

Do not apply a scheduler to a logical interface using a wildcard. For example, if you configure a logical interface (unit) with one parameter, and apply a scheduler map to the interface using a wildcard, the logical interface will not apply the scheduler. The following configuration will commit correctly but will not apply the scheduler map to interface **so-3/0/0.0**:

```
[edit]
user@host# set class-of-service interfaces so-3/0/* unit 0 scheduler-map
  MY_SCHED_MAP
user@host# set class-of-service interfaces so-3/0/0 unit 0 shaping-rate 100m
```

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

```
user@host# show
```

```
interfaces {
  so-3/0/* {
    unit 0 {
      scheduler-map MY_SCHED_MAP;
    }
  }
  so-3/0/0 {
    unit 0 {
      shaping-rate 100m;
    }
  }
}
```

Recommended: Identifying Logical Interfaces Individually When Applying a Scheduler

Always apply the scheduler to a logical interface without the wildcard:

```
[edit]
user@host# set class-of-service interfaces so-3/0/0 unit 0 scheduler-map
  MY_SCHED_MAP
user@host# set class-of-service interfaces so-3/0/0 unit 0 shaping-rate 100m
```

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

```
user@host# show
```


Copyright © 2019, Juniper Networks, Inc.

```
interfaces {
  so-3/0/0 {
    unit 0 {
      scheduler-map MY_SCHED_MAP;
      shaping-rate 100m;
    }
  }
}
```

NOTE: This same wildcard behavior applies to classifiers and rewrites as well as schedulers.

Related Documentation

- Configuring Scheduler Maps on page 258
- Applying Scheduler Maps to Physical Interfaces on page 259
- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300

Configuring a Policer Overhead

Configuring a policer overhead allows you to control the rate of traffic sent or received on an interface. When you configure a policer overhead, the configured policer overhead value (bytes) is added to the length of the final Ethernet frame. This calculated length of frame is used to determine the policer or the rate limit action. Therefore, the policer overhead enables you to control the rate of traffic sent or received on an interface. You can configure the policer overhead to rate-limit queues and Layer 2 and MAC policers. The policer overhead and the shaping overhead can be configured simultaneously on an interface.

This feature is supported on M Series and T Series routers with IQ2 PICs or IQ2E PICs, and on MX Series DPCs.

To configure a policer overhead for controlling the rate of traffic sent or received on an interface:

1. In the **[edit chassis]** hierarchy level in configuration mode, create the interface on which to add the policer overhead to input or output traffic.

   ```
   [edit chassis]
   user@host# edit fpc fpc pic pic
   ```

   For example:

   ```
   [edit chassis]
   user@host# edit fpc 0 pic 1
   ```

2. Configure the policer overhead to control the input or output traffic on the interface. You could use either statement or both the statements for this configuration.

Copyright © 2019, Juniper Networks, Inc.

```
[edit chassis fpc fpc pic pic]
user@host# set ingress-policer-overhead bytes;
user@host# set egress-policer-overhead bytes;
```

For example:

```
[edit chassis fpc 0 pic 1]
user@host# set ingress-policer-overhead 10;
user@host# set egress-policer-overhead 20;
```

3. Verify the configuration:

```
[edit chassis]
user@host# show
fpc 0 {
    pic 1 {
        ingress-policer-overhead 10;
        egress-policer-overhead 20;
    }
}
```

NOTE: When the configuration for the policer overhead bytes on a PIC is changed, the PIC goes offline and then comes back online. In addition, the configuration in the CLI is on a per-PIC basis and, therefore, applies to all the ports on the PIC.

Related Documentation

- egress-policer-overhead on page 1097
- ingress-policer-overhead on page 1167

CoS for L2TP Tunnels on Ethernet Interface Overview

For effective packet tunneling, CoS is implemented over L2TP tunnels. For Ethernet interfaces, CoS is supported for L2TP session traffic to a LAC on platforms configured as an LNS that include egress IQ2 or IQ2E PICs.

This feature is supported on the following platforms:

- EX Series switches
- M7i and M10i routers
- M120 routers

To enable session-aware CoS on an L2TP interface, include the **per-session-scheduler** statement at the **[edit interfaces unit *logical-unit-number*]** hierarchy level.

Copyright © 2019, Juniper Networks, Inc.

After CoS is configured on an L2TP tunnel, Junos OS dynamically creates a traffic shaper for the traffic-shaping-profile and the L2TP tunnel based on the tunnel identification number. This ensures that the packets are monitored at the LAC and classified to allow the traffic flow to be adjusted on congested networks.

This feature has the following limitations:

- Only 991 shapers are supported on each IQ2 or IQ2E PIC.
- For a 4-port IQ2E PIC, you can configure up to 1976 shapers for an 8-queue session and 3952 shapers for a 4-queue session.
- For an 8-port IQ2E PIC, you can configure up to 1912 shapers for an 8-queue session and up to 3824 shapers for a 4-queue session.
- Sessions in excess of the maximum supported values specified for the PICs cannot be shaped (but they can be policed).
- There is no support for PPP multilinks.
- The overall traffic rate cannot exceed the L2TP traffic rate, or else random drops result.
- There is no support for logical interface scheduling and shaping at the ingress because all schedulers are now reserved for L2TP.
- There is no support for physical interface rate shaping at the ingress.
- You cannot delete or deactivate the primary Ethernet interface on which the tunnel is established.

You can provide policing support for sessions with more than the maximum supported value on each IQ2 or IQ2E PIC. Each session can have four or eight different classes of traffic (queues). Each class needs its own policer; for example, one for voice and one for data traffic.

Related Documentation

- Configuring CoS for L2TP Tunnels on Ethernet Interfaces on page 819
- Configuring LNS CoS for Link Redundancy on page 820
- Example: Configuring L2TP LNS CoS Support for Link Redundancy on page 821

Configuring CoS for L2TP Tunnels on Ethernet Interfaces

The Layer 2 Tunneling Protocol (L2TP) is often used to carry traffic securely between an L2TP Network Server (LNS) to an L2TP Access Concentrator (LAC). CoS is supported for L2TP session traffic to a LAC on platforms configured as an LNS that include egress IQ2 and IQ2E Ethernet PICs.

This feature is supported on the following platforms:

- EX Series switches
- M7i and M10i routers
- M120 routers

Copyright © 2019, Juniper Networks, Inc.

To configure CoS for L2TP on Ethernet interfaces:

1. Configure L2TP services on the Ethernet interface.
2. On the Ethernet interface, enable session-aware CoS for L2TP sessions.

```
[[edit interfaces interface-name  unit logical-unit-number]
user@host# set per-session-scheduler
```

3. Configure the traffic manager in the IQ2 or IQ2E PIC to enable per-session CoS support.

```
[edit chassis fpc slot-number pic pic-number]
user@host# set traffic-manager mode-session-shaping
```

4. (Optional) To fine tune the system, you may also set the traffic-manager mode to session-shaping and configure the value of ingress-shaping-overhead parameter from 50 through 130 depending on your network requirement.

```
[edit chassis fpc slot-number pic pic-number]
user@host# set traffic-manager ingress-shaping-overhead value mode-session-shaping
```

NOTE: If you deactivate or delete the primary Ethernet interface on which the L2TP tunnel is configured, the tunnel with sessions having CoS is torn down.

After CoS is enabled for L2TP tunnels on Ethernet interface, you can run the **show class-of-service l2tp-session** command to verify the mapping of CoS with the configured L2TP session.

Related Documentation

- *L2TP Minimum Configuration*
- Configuring CoS for L2TP Tunnels on ATM Interfaces on page 848
- CoS for L2TP Tunnels on Ethernet Interface Overview on page 818
- Configuring LNS CoS for Link Redundancy on page 820
- Example: Configuring L2TP LNS CoS Support for Link Redundancy on page 821
- **show class-of-service l2tp-session on page 1397**

Configuring LNS CoS for Link Redundancy

You can configure multiple ports on the same IQ2 and IQ2E PICs to support link redundancy for CoS on L2TP tunnels configured on an Ethernet interface. Link redundancy is useful when the active port is unavailable due to events such as:

- Disconnection of the cable
- Rebooting of the remote end system
- Traffic re-routing through a different port due to network conditions

Copyright © 2019, Juniper Networks, Inc.

When link redundancy is enabled in such scenarios, the L2TP tunnels and its session are maintained by switching traffic to another port configured on the same IQ2 or IQ2E PIC.

To configure multiple ports (IQ and IQ2PE PIC) on an Ethernet interface for redundancy with CoS, configure per-session-scheduler for all Ethernet ports:

```
user@host#edit interfaces ge-2/0/0 unit 0 per-session-scheduler
user@host#edit interfaces ge-2/0/1 unit 0 per-session-scheduler
```

You can similarly configure all the ports on the IQ2 or IQ2E PIC to support link redundancy for CoS on L2TP tunnels.

NOTE:

- If one or more redundancy ports is removed from the configuration, the tunnels established through those redundancy ports also go down.
- You must configure per-session-scheduler for all the ports that are to be used for redundancy. If you do not do so, new tunnels or sessions with CoS do not get established.

Related Documentation

- per-session-scheduler on page 1225

Example: Configuring L2TP LNS CoS Support for Link Redundancy

This example shows how link redundancy is supported when CoS for L2TP is configured on Ethernet interfaces.

NOTE: In this example, support for link redundancy is demonstrated by manually disabling the interface. However, link redundancy is also supported when the interface goes down due to events such as disconnection of the cable or rebooting of the remote end system.

- Requirements on page 821
- Overview on page 822
- Configuration on page 822
- Verification on page 823

Requirements

Before you begin:

- Configure service and loopback interfaces.
- Configure CoS for L2TP.

Copyright © 2019, Juniper Networks, Inc.

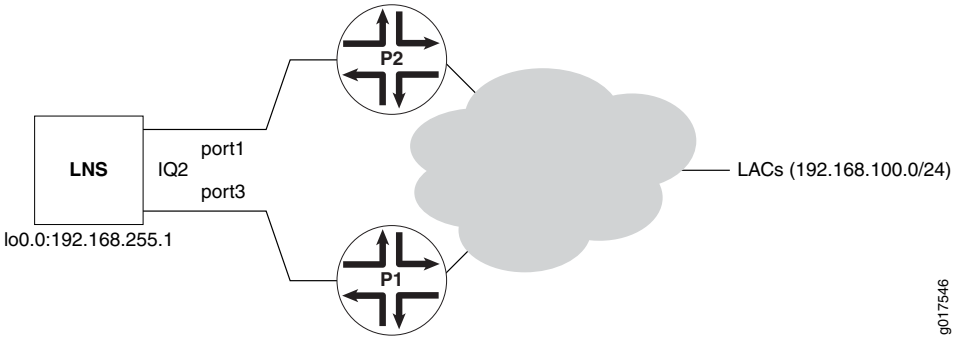
This feature applies to M Series Multiservice Edge Router running Junos OS Release 12.1 or later and EX Series switches.

Overview

Junos OS now supports link redundancy for CoS configured on an L2TP LNS. In this example, we verify that an L2TP tunnel does not go down when the Ethernet interface, through which the tunnels and its sessions with CoS are established, goes down.

Figure 63 on page 822 shows a sample scenario in which L2TP access concentrator (LAC) devices operate on one side of an L2TP tunnel. LAC devices are configured with the address range of 192.168.100.0 with a subnet mask of 24. The LAC devices are connected to two backbone routers, P1 and P2. These two routers, P1 and P2, are connected over two Gigabit Ethernet ports on a single Ethernet IQ2 PIC to an L2TP network server (LNS). The LNS device is a router running Junos OS that supports redundancy for terminating L2TP sessions configured with CoS parameters. The CoS settings are applied on the interfaces using a RADIUS server when the L2TP session is set up. One of the Gigabit Ethernet interfaces on the IQ2 PIC present on the LNS device, ge-0/3/1, is connected to P1, while the other interface, ge-0/3/3, is linked to P2. Such a method of connection enables the subscriber sessions that reach the LAC devices to be forwarded to one of the two ports of the IQ2 PIC on the LNS device.

Figure 63: Topology to Verify Link Redundancy Support for L2TP LNS CoS



Configuration

Step-by-Step Procedure

To configure Ethernet interfaces for redundancy:

1. Configure Gigabit Ethernet interfaces.

```
[edit interfaces]
user@host# set ge-0/3/1 unit 0 family inet address 192.168.1.1/30
user@host# set ge-0/3/3 unit 0 family inet address 192.168.1.5/30
user@host# set ge-0/3/1 unit 0 per-session-scheduler
user@host# set ge-0/3/3 unit 0 per-session-scheduler
```

2. Configure static routing options.

```
[edit routing-options]
user@host# set static route 192.168.100.0/24 next-hop [ 192.168.1.2 192.168.1.6 ]
```


Copyright © 2019, Juniper Networks, Inc.

Step-by-Step Procedure

Verify that CoS is now implemented over L2TP on an Ethernet interface and the LAC is reachable.

1. Verify that LAC is reachable.

```
user@host> show route 192.168.100.1
```

```
inet.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.100.0/24   *[Static/5] 1d 02:09:09
                      to 192.168.1.2 via ge-0/3/1.0
                    > to 192.168.1.6 via ge-0/3/3.0
```

2. Bring up an L2TP session and verify that L2TP sessions come up.

```
user@host> show services l2tp session
```

```
Interface: sp-1/3/0, Tunnel group: GEN-TUN-GRP-BIO, Tunnel local ID: 44806
  Local Remote Interface State              Bundle Username
  ID    ID     unit
  12491  33795         1 Established             - test1
```

3. Send a traffic stream towards the subscriber.

4. Verify that the shaping at the subscriber end is as per the shaping rate configured.

```
user@host# show class-of-service l2tp-session
```

```
L2TP Session Username: test1, Index: 12491
Physical interface: ge-0/3/3, Index: 131
Queues supported: 4, Queues in use: 4
  Scheduler map: GEN-SCHED-MAP-EF-65%, Index: 5212
  Shaping rate: 2162200 bps
  Encapsulation Overhead: 6, Cell Overhead: Enabled
```

In the output of the **show class-of-service l2tp-session** command, ge-0/3/3, index 131 represents the port used to establish the L2TP tunnel to which the current L2TP session belongs. It does not represent the port that was active when the L2TP session came up.

Verification

Verify that, when CoS is configured on an L2TP tunnel, link redundancy works if one of the ports on which the L2TP tunnel is established goes down.

- Bring Down ge-0/3/3 Interface Through Which the L2TP Tunnel Is Established on page 823
- Verify LAC Reachability and the Status of L2TP Sessions on page 824

Bring Down ge-0/3/3 Interface Through Which the L2TP Tunnel Is Established

Purpose Bring down the interface through which the L2TP session and its tunnels are established.

Copyright © 2019, Juniper Networks, Inc.

Action

```
[edit interfaces]
user@host# set ge-0/3/3 disable
user@host# commit
```

Verify LAC Reachability and the Status of L2TP Sessions

Purpose Verify that link redundancy works and the L2TP session does not go down when the active port on the IQ2 PIC is down. Verify that the traffic flow is unaffected after it is switched to another port configured on the same IQ2 or IQ2E PIC.

Action

```
user@host> show route 192.168.100.1
inet.0: 14 destinations, 14 routes (14 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.100.0/24   *[Static/5] 1d 02:35:09
                       to 192.168.1.2 via ge-0/3/1.0
```

```
user@host> show services l2tp session
Interface: sp-1/3/0, Tunnel group: GEN-TUN-GRP-BIO, Tunnel local ID: 44806
  Local Remote Interface State               Bundle Username
  ID    ID     unit
  12491  33795         1 Established              - test1
```

Related Documentation

- Configuring LNS CoS for Link Redundancy on page 820

Configuring Shaping on 10-Gigabit Ethernet IQ2 PICs

The 10-Gigabit Ethernet IQ2 PIC (which has **xe-** interfaces) is unlike other Gigabit Ethernet IQ2 PICs in that it does not have oversubscription. The bandwidth from the PIC to the FPC is sufficient to transmit the full line rate. However, the 10-Gigabit Ethernet IQ2 PIC has the same hardware architecture as other Gigabit Ethernet IQ2 PICs and supports all the same class-of-service (CoS) features. For more information, see the Ethernet Interfaces Feature Guide for Routing Devices.

To handle oversubscribed traffic on 10-Gigabit Ethernet IQ2 PICs, you can configure input shaping and scheduling based on Layer 2, MPLS, and Layer 3 packet fields. Gigabit Ethernet IQ2 PICs also support simple filters, accounting, and policing. This topic discusses input and output shaping and scheduling. For information about accounting and policing, see the *Junos OS Network Interfaces Library for Routing Devices*.

Copyright © 2019, Juniper Networks, Inc.

NOTE: The CoS functionality supported on Gigabit Ethernet IQ2 PICs is not available across aggregated Ethernet links. However, if you configure a CoS scheduler map on the link bundle, the configuration is honored by the individual links within that bundle.

Therefore, CoS behaves as configured on a per-link level, but *not* across the aggregated links.

If you configure a shaping transmit rate of 100 Mbps on an aggregated Ethernet bundle with three ports (by applying a scheduler for which the configuration includes the **transmit-rate** statement with the **exact** option at the **[edit class-of-service schedulers** *scheduler-name*] hierarchy level), each port is provisioned with a 33.33 Mbps shaping transmit rate.

You can configure shaping for aggregated Ethernet interfaces that use interfaces originating from Gigabit Ethernet IQ2 PICs. However, you cannot enable shaping on aggregated Ethernet interfaces when the aggregate bundle combines ports from IQ and IQ2 PICs.

By default, transmission scheduling is not enabled on logical interfaces. Logical interfaces without shaping configured share a default scheduler. This scheduler has a committed information rate (CIR) that equals 0. (The CIR is the guaranteed rate.) The default scheduler has a peak information rate (PIR) that equals the physical interface shaping rate. The default operation can be changed by configuring the software.

To configure input and output scheduling and shaping on 10-Gigabit Ethernet IQ2 PICs:

1. Create the traffic-control profile, including the required scheduler map and shaping rate.

```
[edit class-of-service]
user@host# edit traffic-control-profiles tcp1
   user@host# set scheduler-map map1
user@host# set shaping-rate 100
```

2. Apply the traffic-control profile to logical interface for either input scheduling and shaping and/or output scheduling and shaping:

   To apply the traffic-control profile to a logical interface for input scheduling and shaping.

```
[edit class-of-service]
user@host# edit interfaces xe-1/2/1 unit 0
user@host# set input-traffic-control-profile tcp1 shared-instance map1
```

   To apply the traffic-control profile to a logical interface for output scheduling and shaping.

```
user@host# set output-traffic-control-profile tcp1 shared-instance map1
```

Copyright © 2019, Juniper Networks, Inc.

To apply the traffic-control profile to a logical interface for both input and output scheduling and shaping.

```
[edit class-of-service]
user@host# edit interfaces xe-1/2/1 unit 0
user@host# set input-traffic-control-profile tcp1 shared-instance map1
user@host# set output-traffic-control-profile tcp1 shared-instance map1
```

3. Enable per-unit scheduling (per-unit-scheduler statement) to enable the association of scheduler maps with logical interfaces.

```
[edit interfaces xe-1/2/1
user@host# set per-unit-scheduler
```

NOTE: The **scheduler-map** and **shaping-rate** statements can be specified at the **[edit class-of-service interfaces** *interface-name* **unit** *logical-unit-number***]** hierarchy level. However, we do not recommend this configuration. Include the **output-traffic-control-profile** or **input-traffic-control-profile** statement instead.

NOTE: For Gigabit Ethernet IQ2 interfaces, the logical interface egress statistics displayed in the **show interfaces** command output might not accurately reflect the traffic on the wire when output shaping is applied. Traffic management output shaping might drop packets after they are tallied by the **Output bytes** and **Output packets** logical interface counters. However, correct values display for both of these **Transit statistics** when per-unit scheduling is enabled for the Gigabit Ethernet IQ2 physical interface, or when a single logical interface is actively using a shared scheduler.

Related Documentation

- *Simple Filter Overview*
- *How Simple Filters Evaluate Packets*
- *Guidelines for Configuring Simple Filters*
- *Guidelines for Applying Simple Filters*

Configuring Per-Unit Scheduling for GRE Tunnels Using IQ2 and IQ2E PICs

This topic describes how to configure per unit scheduling for Generic Routing Encapsulation (GRE) tunnels running on Intelligent Queuing 2 (IQ2) PICs and Intelligent Queuing 2 Enhanced (IQ2E) PICs in M7i, M10i, M120, M320, T Series and TX Matrix routers.

M7i, M10i, M120, M320, T Series and TX Matrix routers with Intelligent Queuing 2 (IQ2) PICs and Intelligent Queuing 2 Enhanced (IQ2E) PICs support per unit scheduling for

Copyright © 2019, Juniper Networks, Inc.

Generic Routing Encapsulation (GRE) tunnels, adding all the functionality of tunnel PICs to GRE tunnels. The class of service (CoS) for the GRE tunnel traffic is applied as the traffic is looped through the IQ2 and IQ2E PIC.

Shaping is performed on full packets that pass through the GRE tunnel.

IQ2 and IQ2E PICs support all interfaces that are supported on tunnel PICs, as follows:

- **gr-*fpc/pic/port***
- **vt-*fpc/pic/port***
- **lt-*fpc/pic/port***
- **ip-*fpc/pic/port***
- **pe-*fpc/pic/port***
- **pd-*fpc/pic/port***
- **mt-*fpc/pic/port***

The port variable is always zero.

The IQ2 and IQ2E PICs tunnel functionality is the same as that of regular tunnel PICs.

You can specify that IQ2 and IQ2E PICs work exclusively in tunnel mode or as a regular PIC. When IQ2 and IQ2E PICs work exclusively as a tunnel PIC, they support the same number of tunnel logical interfaces as regular tunnel PICs; for example each PIC can support 4,000 **gr-** logical interfaces. The default setting uses IQ2 and IQ2E PICs as a regular PIC.

IQ2E PIC schedulers can be dynamically allocated across ports.

NOTE: When IQ2 and IQ2E PICs work exclusively in tunnel mode only **traffic-control-profile** on **gr-** logical interfaces are supported. Class of Service (CoS) on **gr-** logical interfaces is not supported.

Also, a scheduler is allocated for a **gr-** logical interface only when there is a traffic-control profile configured for it.

The **gr-** logical interfaces without an explicit CoS configuration are not assigned a dedicated scheduler. These use a reserved scheduler meant for all unshaped tunnel traffic; that is, all traffic on **gr-** logical interfaces that do not have CoS configured and all traffic from other types of tunnels.

On **gr-** interfaces, you can configure an output traffic control profile on the logical interface:

1. Configuring the IQ2 and IQ2E PIC to work exclusively in tunnel mode

   For example:

```
[edit]
user@host# set chassis fpc 1 pic 1 tunnel-services tunnel-only
```

Copyright © 2019, Juniper Networks, Inc.

NOTE: The PIC will be automatically bounced when the tunnel services configuration is changed.

The **chassis traffic-manager** mode must have the ingress traffic manager enabled in order for the tunnel-services to work correctly.

2. Create the traffic control profile and specify the shaping rate.

   For example:

```
[edit]
user@host# set class-of-service traffic-control-profiles tcp shaping-rate 1000
user@host# set class-of-service interfaces gr-1/1/1 unit 0 output-traffic-control-profile
  tcp
```

3. Apply the profile to the logical interface as an output traffic control profile.

   For example:

```
[edit]
user@host# set class-of-service interfaces gr-1/1/1 unit 0 output-traffic-control-profile
  tcp
```

4. To verify the configuration and view statistics:

   - You can use the **show interfaces queue gr-*fpc/pic/port*** command to display statistics for the specified tunnel.
   - To view the configuration and statistics for GRE tunnel logical interfaces, use the **show interfaces queue gr-** command.

Related Documentation

- tunnel-services (Chassis) on page 1316
- Configuring CoS for GRE and IP-IP Tunnels on page 687
- CoS on Enhanced IQ2 PICs Overview on page 794

Understanding Burst Size Configuration on IQ2 and IQ2E Interfaces

You can explicitly configure the burst size for shapers in a traffic control profile for IQ2 and IQ2E interfaces. This feature is supported on M7i, M10i, M40e, M120, M320 routers and all T Series routers.

The shaping burst size determines the maximum number of bytes that can be sent through a shaper during a burst. The guaranteed burst size determines when the scheduler moves from green to yellow.

Copyright © 2019, Juniper Networks, Inc.

The burst size limits the number of credits that can be accumulated for scheduling. Configuring a burst size is only useful in the case when traffic is sent after a long lull period so that credits can be accumulated until the burst size limit is reached. When traffic is continuous, credits are not accumulated, and the burst size limit is not reached.

If no burst size value is specified when the shaping rate or guaranteed rate is configured, then a default burst size (expressed as a time value) is applied. The default shaping burst size is 10 ms of the shaping rate (that is, 10*shaping rate/1000 bytes). The minimum value is 2048 bytes to accommodate the minimum of 1 MTU.

The burst size value is adjusted and rounded off to meet the restrictions enforced by the hardware. Thus, the actual burst size in the hardware might vary slightly from the configured value.

To enable this feature, include the **burst-size** statement at the following hierarchy levels:

```
[edit class-of-service traffic-control-profiles shaping-rate]
[edit class-of-service traffic-control-profiles guaranteed-rate]
```

NOTE: The guaranteed-rate burst size value cannot be greater than the shaping-rate burst size.

Related Documentation

- Configuring Burst Size for Shapers on IQ2 and IQ2E Interfaces on page 829
- guaranteed-rate on page 1149
- shaping-rate on page 1269

Configuring Burst Size for Shapers on IQ2 and IQ2E Interfaces

This topic shows how to set the **burst-size** while configuring the **shaping-rate** and **guaranteed-rate** under the **[edit class-of-service traffic-control-profiles *profile-name*]** hierarchy level.

In following configuration for tcp1, the **shaping-rate** burst size is set to 5 KB, and the **guaranteed-rate** burst size is set to 3 KB under the **traffic-control-profiles** statement. To apply this configuration to a logical interface (ifl), the traffic-control-profile is attached to the ifl.

```
class-of-service {
  traffic-control-profiles {
    tcp1 {
      shaping-rate 100m burst-size 5k;
      guaranteed-rate 50m burst-size 3k;
    }
    tcp2 {
      shaping-rate 100m burst-size 5k;
    }
  }
```

Copyright © 2019, Juniper Networks, Inc.

```
  interfaces {
    interface-set ifset1 {
      output-traffic-control-profile tcp1;
    }
    ge-1/2/1 {
      unit 0 {
        output-traffic-control-profile tcp1;
      }
    }
    ge-1/2/2 {
      output-traffic-control-profile tcp2;
    }
  }
}
```

Related Documentation

- Understanding Burst Size Configuration on IQ2 and IQ2E Interfaces on page 828
- guaranteed-rate on page 1149
- shaping-rate on page 1269

Configuring a CIR and a PIR on Ethernet IQ2 Interfaces

On Gigabit Ethernet IQ2 interfaces, you can configure a CIR (guaranteed rate) and a PIR (shaping rate) on a single logical interface. The configured rates are gathered into a traffic control profile. If you configure a traffic control profile with a CIR (guaranteed rate) only, the PIR (shaping rate) is set to the physical interface (port) rate.

In the following example, logical unit **0** has a CIR equal to 30 Mbps and a PIR equal to 200 Mbps. Logical unit **1** has a PIR equal to 300 Mbps. Logical unit **2** has a CIR equal to 100 Mbps and a PIR that is unspecified. For logical unit **2**, the software causes the PIR to be 1 Gbps (equal to the physical interface rate) because the PIR must be equal to or greater than the CIR.

Excess bandwidth is the leftover bandwidth on the port after meeting all the guaranteed rate requirements of the logical interfaces. For each port, excess bandwidth is shared as follows:

- Proportional to the guaranteed rate—This method is used if you configure one or more logical interfaces on a port to have a guaranteed rate.
- Proportional to the shaping rate—This method is used if you configure none of the logical interfaces on a port to have a guaranteed rate.

In this example, bandwidth is shared proportionally to the guaranteed rate because at least one logical interface has a guaranteed rate.

1. Create and configure the traffic control profile.

```
[edit]
user@host# set class-of-service traffic-control-profiles profile1 scheduler-map
  sched-map
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set class-of-service traffic-control-profiles profile1 shaping-rate 200m
user@host# set class-of-service traffic-control-profiles profile1 guaranteed-rate 30m
user@host# set class-of-service traffic-control-profiles profile1 delay-buffer-rate
  150m
user@host# set class-of-service traffic-control-profiles profile2 scheduler-map
  sched-map
user@host# set class-of-service traffic-control-profiles profile2 shaping-rate 300m
user@host# set class-of-service traffic-control-profiles profile2 delay-buffer-rate
  500k
user@host# set class-of-service traffic-control-profiles profile3 scheduler-map
  sched-map
user@host# set class-of-service traffic-control-profiles profile3 guaranteed-rate 100m
```

2. Apply the traffic control profiles to the interfaces

```
[edit]
user@host# set class-of-service interfaces ge-3/0/0 unit 0
  output-traffic-control-profile profile1
user@host# set class-of-service interfaces ge-3/0/0 unit 1
  output-traffic-control-profile profile2
user@host# set class-of-service interfaces ge-3/0/0 unit 2
  output-traffic-control-profile profile3
```

3. View the configuration.

```
[edit]
```

```
user@host# show class-of-service traffic-control-profiles
```

```
profile1 {
    scheduler-map sched-map;
    shaping-rate 200m;
    guaranteed-rate 30m;
    delay-buffer-rate 150m;
}
profile2 {
    scheduler-map sched-map;
    shaping-rate 300m;
    delay-buffer-rate 500k;
}
profile3 {
    scheduler-map sched-map;
    guaranteed-rate 100m;
}
```

```
[edit]
```

```
user@host# show class-of-service interfaces
```

```
ge-3/0/0 {
    unit 0 {
        output-traffic-control-profile profile1;
    }
```

Copyright © 2019, Juniper Networks, Inc.

```
    unit 1 {
        output-traffic-control-profile profile2;
    }
    unit 2 {
        output-traffic-control-profile profile3;
    }
}
```

Related Documentation

- CoS on Enhanced IQ2 PICs Overview on page 794
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

Example: Configuring Shared Resources on Ethernet IQ2 Interfaces

For input traffic on physical interface **ge-1/2/3**, logical interface units **1**, **2**, and **3** are sharing one set of scheduler-shaper resources, defined by traffic-control profile **s1**. Logical interface units **4**, **5**, and **6** are sharing another set of scheduler-shaper resources, defined by traffic-control profile **s1**.

For output traffic on physical interface **ge-1/2/3**, logical interface units **1**, **2**, and **3** are sharing one set of scheduler-shaper resources, defined by traffic-control profile **s2**. Logical interface units **4**, **5**, and **6** are sharing another set scheduler-shaper resources, defined by traffic-control profile **s2**.

For each physical interface, the **shared-instance** statement creates one set of resources to be shared among units **1**, **2**, and **3** and another set of resources to be shared among units **4**, **5**, and **6**. Input and output shaping rates are configured at the physical interface level, which demonstrates the hierarchical shaping capability of the Gigabit Ethernet IQ2 PIC.

```
[edit]
class-of-service {
  traffic-control-profiles {
    s1 {
      scheduler-map map1;
      shaping-rate 100k;
    }
    s2 {
      scheduler-map map1;
      shaping-rate 200k;
    }
  }
  forwarding-classes { # Map one forwarding class to one queue.
    queue 0 fc-be;
    queue 1 fc-be1;
    queue 2 fc-ef;
    queue 3 fc-ef1;
    queue 4 fc-af11;
    queue 5 fc-af12;
    queue 6 fc-nc1;
    queue 7 fc-nc2;
```


Copyright © 2019, Juniper Networks, Inc.

```
}
classifiers { # Map 802.1p bits to forwarding-class and loss-priority.
  ieee-802.1 ieee-8021p-table {
    forwarding-class fc-nc2 {
      loss-priority low code-points [111];
    }
    forwarding-class fc-nc1 {
      loss-priority low code-points [110];
    }
    forwarding-class fc-af12 {
      loss-priority low code-points [101];
    }
    forwarding-class fc-af11 {
      loss-priority low code-points [100];
    }
    forwarding-class fc-ef1 {
      loss-priority low code-points [011];
    }
    forwarding-class fc-ef {
      loss-priority low code-points [010];
    }
    forwarding-class fc-be1 {
      loss-priority low code-points [001];
    }
    forwarding-class fc-be {
      loss-priority low code-points [000];
    }
  }
}
interfaces {
  ge-1/2/3 {
    input-shaping-rate 500m;
    shaping-rate 500m; # Output shaping rate
    unit 0 { # Apply behavior aggregate classifier to an interface.
      classifiers {
        ieee-802.1 ieee-8021p-table;
      }
    }
    unit 1 {
      input-traffic-control-profile s1 shared-instance 1;
      output-traffic-control-profile s2 shared-instance 1;
    }
    unit 2 {
      input-traffic-control-profile s1 shared-instance 1;
      output-traffic-control-profile s2 shared-instance 1;
    }
    unit 3 {
      input-traffic-control-profile s1 shared-instance 1;
      output-traffic-control-profile s2 shared-instance 1;
    }
    unit 4 {
      input-traffic-control-profile s1 shared-instance 2;
      output-traffic-control-profile s2 shared-instance 2;
    }
    unit 5 {
```

```
            input-traffic-control-profile s1 shared-instance 2;
            output-traffic-control-profile s2 shared-instance 2;
        }
        unit 6 {
            input-traffic-control-profile s1 shared-instance 2;
            output-traffic-control-profile s2 shared-instance 2;
        }
    }
  }
}
```

Configuring a Simple Filter

Configure a simple filter that overrides the classification derived from the lookup of the Layer 2 fields.

```
firewall {
  family inet {
    simple-filter sf-1 {
      term 1 {
        source-address 172.16.0.0/24;
        destination-address 172.16.20.0/24;
        source-port 1024-9071;
      }
      then { # Action with term-1
      forwarding-class fc-be1;
      loss-priority high;
    }
    term 2 {
      source-address 172.16.10.0/24;
      destination-address 172.16.30.0/24;
    }
    then { # Action with term-2
    forwarding-class fc-ef1;
    loss-priority low;
  }
}
interfaces { # Apply the simple filter.
ge-1/2/3 {
  unit 0 {
    family inet {
      simple-filter {
        input sf-1;
      }
    }
  }
}

class-of-service {
  scheduler-maps { # Configure a custom scheduler map.
  map1 {
    forwarding-class fc-be scheduler sch-Q0;
    forwarding-class fc-be1 scheduler sch-Q1;
    forwarding-class fc-ef scheduler sch-Q2;
    forwarding-class fc-ef1 scheduler sch-Q3;
    forwarding-class fc-af11 scheduler sch-Q4;
```


Copyright © 2019, Juniper Networks, Inc.

```
      forwarding-class fc-af12 scheduler sch-Q5;
      forwarding-class fc-nc1 scheduler sch-Q6;
      forwarding-class fc-nc2 scheduler sch-Q7;
    }
}
schedulers { # Define schedulers.
sch-Q0 {
   transmit-rate percent 25;
   buffer-size percent 25;
   priority low;
   drop-profile-map loss-priority any protocol any drop-profile drop-default;
}
sch-Q1 {
   transmit-rate percent 5;
   buffer-size temporal 2000;
   priority high;
   drop-profile-map loss-priority any protocol any drop-profile drop-ef;
}
sch-Q2 {
   transmit-rate percent 35;
   buffer-size percent 35;
   priority low;
   drop-profile-map loss-priority any protocol any drop-profile drop-default;
}
sch-Q3 {
   transmit-rate percent 5;
   buffer-size percent 5;
   drop-profile-map loss-priority any protocol any drop-profile drop-default;
}
sch-Q4 {
   transmit-rate percent 5;
   priority high;
   drop-profile-map loss-priority any protocol any drop-profile drop-ef;
}
sch-Q5 {
   transmit-rate percent 10;
   priority high;
   drop-profile-map loss-priority any protocol any drop-profile drop-ef;
}
sch-Q6 {
   transmit-rate remainder;
   priority low;
   drop-profile-map loss-priority any protocol any drop-profile drop-default;
}
sch-Q7 {
   transmit-rate percent 5;
   priority high;
   drop-profile-map loss-priority any protocol any drop-profile drop-default;
}
```

Related Documentation

- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

Copyright © 2019, Juniper Networks, Inc.

Configuring and Applying IEEE 802.1ad Classifiers

If you apply an IEEE 802.1 classifier to a logical interface, this classifier takes precedence and is not compatible with any other classifier type. For Juniper Networks MX Series 5G Universal Routing Platform interfaces or IQ2 PICs with IEEE 802.1ad frame formats or EX Series switches, you can set the forwarding class and loss priority for traffic on the basis of the three IEEE 802.1p bits (three bits in either the inner virtual LAN (VLAN) tag or the outer VLAN tag) and the drop eligible indicator (DEI) bit. You can apply the default map or customize one or more of the default values.

You then apply the classifier to the interface on which you configure IEEE 802.1ad frame formats.

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

1. Define the custom IEEE 802.1ad map:

   a. Create the classifier by specifying a name for it and defining it as an IEEE-802.1ad (DEI) classifier.

      ```
      [edit]
      user@host# edit class-of-service classifiers ieee-802.1ad dot1p_dei_class
      ```

   b. Assign the forwarding class and loss priority to the code-point alias.

      ```
      [edit class-of-service classifiers ieee-802.1ad dot1p_dei_class]
      user@host# set forwarding-class best-effort loss-priority low code-points [0000
        1101]
      ```

2. Apply the classifier to the logical interface:

   a. Specify the interface to which you want to apply the classifier.

      ```
      [edit]
      user@host# edit class-of-service interfaces ge-2/0/0 unit 0
      ```

   b. Specify the name of the classifier you want to apply to the interface.

      ```
      [edit class-of-service interfaces ge-2/0/0 unit 0]
      user@host# set classifiers ieee-802.1ad dot1p_dei_class
      ```

3. Verify the custom IEEE 802.1ad map configuration:

   ```
   [edit]
   user@host# show
   ```

   ```
   class-of-service {
     classifiers {
       ieee-802.1ad dot1p_dei_class {
         forwarding-class best-effort {
   ```


Copyright © 2019, Juniper Networks, Inc.

```
            loss-priority low code-points [ 0000 1101 ];
        }
    }
  }
}
```

```
class-of-service {
  interfaces {
  ge-2/0/0 {
    unit 0 {
      classifiers {
        ieee-802.1ad dot1p_dei_class;
      }
    }
  }
  }
]
```

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53

Configuring Rate Limits to Protect Lower Queues on IQ2 and Enhanced IQ2 PICs

You can rate-limit strict-high and high priority queues on IQ2 and IQ2E PICs. Without this limiting, traffic in higher priority queues can block the transmission of lower priority packets. Unless limited, higher priority traffic is always sent before lower priority traffic, causing the lower priority queues to "starve," which in turn leads to timeouts and unnecessary resending of packets.

On the IQ2 and IQ2E PICs you can rate-limit queues before the packets are queued for output. All packets exceeding the configured rate limit are dropped, so care is required when establishing this limit.

NOTE: IQ2E PICs exclude the transmit rate of strict-high and high priority queues, thereby allowing low and medium priority queues to be configured up to 100 percent.

To rate-limit queues, include the **transmit-rate** statement with the **rate-limit** option at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level:

```
[edit class-of-service schedulers scheduler-name]
transmit-rate rate rate-limit;
```

Copyright © 2019, Juniper Networks, Inc.

This example limits the transmit rate of a strict-high expedited-forwarding queue to 1 megabit per second (Mbps). The scheduler and scheduler map are defined and then applied to the traffic at the **[edit interfaces]** and **[edit class-of-service]** hierarchy levels:

```
[edit class-of-service]
schedulers {
   scheduler-1 {
      transmit-rate 1m rate-limit; # This establishes the limit
      priority strict-high;
   }
}
scheduler-maps {
   scheduler-map-1 {
      forwarding-class expedited-forwarding scheduler scheduler-1;
   }
}

[edit interfaces]
s0-2/1/0 {
   per-unit-scheduler;
   encapsulation frame-relay;
   unit 0 {
      dlci 1;
   }
}

[edit class-of-service]
interfaces {
   so-2/1/0 {
      unit 0 {
         scheduler-map scheduler-map-1;
         shaping-rate 2m;
      }
   }
}
```

You can issue the following operational mode commands to verify your configuration (the first shows the rate limit in effect):

- **show class-of-service scheduler-map** *scheduler-map-name*
- **show class-of-service interface** *interface-name*

Related Documentation

- CoS on Enhanced IQ2 PICs Overview on page 794
- Configuring Schedulers on page 258

Copyright © 2019, Juniper Networks, Inc.

Simple Filters Overview

Simple filters are recommended for metropolitan Ethernet applications. They are supported on Gigabit Ethernet intelligent queuing 2 (IQ2) and Enhanced Queuing Dense Port Concentrator (DPC) interfaces only.

Unlike normal filters, simple filters are for IPv4 traffic only and have the following restrictions:

- The **next term** action is not supported.
- Qualifiers, such as the **except** and **protocol-except** statements, are not supported.
- Noncontiguous masks are not supported.
- Multiple source addresses and destination addresses in a single term are not supported. If you configure multiple addresses, only the last one is used.
- Ranges are only valid as source or destination ports. For example, **source-port 400-500** or **destination-port 600-700**.
- Output filters are not supported. You can apply a simple filter to ingress traffic only.
- Simple filters are not supported for interfaces in an aggregated-Ethernet bundle.
- Explicitly configurable terminating actions, such as **accept**, **reject**, and **discard**, are not supported. Simple filters always accept packets.

NOTE: On the MX Series routers with the Enhanced Queuing DPC, the forwarding class is not supported as a from match condition.

Related Documentation

- Configuring a Simple Filter on page 839

Configuring a Simple Filter

This simple filter sets the loss priority to low for TCP traffic with source address **10.1.1.1**, sets the loss priority to high for HTTP (web) traffic with source addresses in the **203.0.113.0/24** range, and sets the loss priority to low for all traffic with destination address **10.6.6.6**. The simple filter is applied as an input filter (arriving packets are checking for destination address **10.6.6.6**, not queued output packets) on interface **ge-0/0/1.0**.

```
[edit]
firewall {
   family inet {
      simple-filter filter1 {
         term 1 {
            from {
               source-address {
                  10.1.1.1/32;
```

Copyright © 2019, Juniper Networks, Inc.

```
            }
            protocol {
              tcp;
            }
          }
          then loss-priority low;
        }
        term 2 {
          from {
            source-address {
              203.0.113.0/24;
            }
            source-port {
              http;
            }
          }
          then loss-priority high;
        }
        term 3 {
          from {
            destination-address {
              10.6.6.6/32;
            }
          }
          then {
            loss-priority low;
            forwarding-class best-effort;
          }
        }
      }
    }
  }
}
interfaces {
  ge-0/0/1 {
    unit 0 {
      family inet {
        simple-filter {
          input filter1;
        }
        address 10.1.2.3/30;
      }
    }
  }
}
```

Related Documentation

- Simple Filters Overview on page 839

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 22

Configuring Class of Service on ATM Interfaces

- CoS on ATM Interfaces Overview on page 841
- Enabling Eight Queues on ATM Interfaces on page 842
- Copying the Packet Loss Priority to the CLP Bit on ATM Interfaces on page 848
- Configuring CoS for L2TP Tunnels on ATM Interfaces on page 848
- Configuring CoS for ATM2 IQ Virtual Circuit Tunnels on page 850
- Applying IEEE 802.1p BA Classifiers to Ethernet VPLS Over ATM on page 851
- Example: Combining Layer 2 and Layer 3 Classification on the Same ATM Physical Interface on page 852
- Applying Scheduler Maps to ATM Interfaces on page 853
- Configuring ATM Scheduler Support for Ethernet VPLS over ATM Bridged Interfaces on page 855
- Example: Configuring ATM Schedulers for Ethernet VPLS over ATM Bridged Interfaces on page 857
- Applying Scheduler Maps to Logical ATM Interfaces on page 858
- Configuring Linear RED Profiles on ATM Interfaces on page 859
- Configuring Virtual Circuit CoS Mode on ATM Interfaces on page 859

CoS on ATM Interfaces Overview

The ATM2 intelligent queuing (IQ) interface allows multiple IP queues into each virtual circuit (VC). On Juniper Networks M Series Multiservice Edge Routers (except the M320 router), a VC tunnel can support four class-of-service (CoS) queues. On M320 routers and T Series Core Routers, for all ATM2 IQ PICs except the OC48 PIC, a VC tunnel can support eight CoS queues. Within a VC tunnel, the weighted round-robin (WRR) algorithm schedules the cell transmission of each queue. You can configure the queue admission policies, such as early packet discard (EPD) or weighted random early detection (WRED), to control the queue size during congestion.

For information about CoS components that apply generally to all interfaces, see "Understanding How Class of Service Manages Congestion and Controls Service Levels

Copyright © 2019, Juniper Networks, Inc.

in the Network" on page 4. For general information about configuring ATM interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.

To configure ATM2 IQ VC tunnel CoS components, include the following statements at the **[edit interfaces at-*fpc/pic/port*]** hierarchy level:

```
[edit chassis fpc slot-number pic pic-number]
max-queues-per-interface number;

[edit interfaces at-fpc/pic/port]
atm-options {
  linear-red-profiles profile-name {
    high-plp-max-threshold percent;
    low-plp-max-threshold percent;
    queue-depth cells high-plp-threshold percentlow-plp-threshold percent;
  }
  plp-to-clp;
  scheduler-maps map-name {
    forwarding-class class-name {
      epd-threshold cells plp1 cells;
      linear-red-profile profile-name;
      priority (high | low);
      transmit-weight (cells number | percent number);
    }
    vc-cos-mode (alternate | strict);
  }
}
unit logical-unit-number {
  atm-scheduler-map (map-name | default);
  family family {
    address address {
      destination address;
    }
  }
  plp-to-clp;
  shaping {
    (cbr rate | rtvbr peak rate sustained rate burst length  |vbr peak rate sustained rate burst
      length);
  }
  vci vpi-identifier.vci-identifier;
}
```

Enabling Eight Queues on ATM Interfaces

By default, IQ, MPC, and DPC interfaces on M120, T320, T640, T1600, TX Matrix, and TX Matrix Plus routers, and MIC or MPC interfaces on MX Series routers, are restricted to a maximum of four egress queues per interface. You can enable eight egress queues by including the **max-queues-per-interface** statement at the **[edit chassis fpc *slot-number* pic *pic-number*]** hierarchy level:

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


Copyright © 2019, Juniper Networks, Inc.

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

If you include the **max-queues-per-interface** statement, all ports on the PIC use the configured maximum.

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

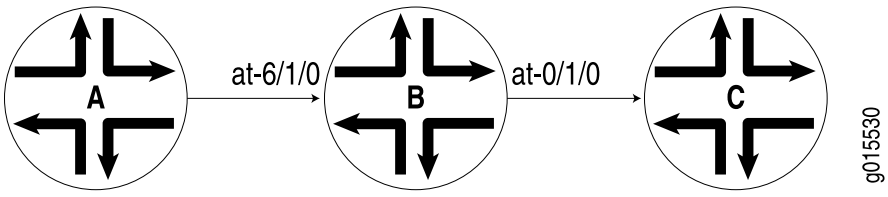
NOTE: When you are considering enabling eight queues on an ATM2 IQ interface, you should note the following:

- ATM2 IQ interfaces using Layer 2 circuit trunk transport mode support only four CoS queues.
- ATM2 IQ interfaces with MLPPP encapsulation support only four CoS queues.
- You can configure only four RED profiles for the eight queues. Thus, queue 0 and queue 4 share a single RED profile, as do queue 1 and queue 5, queue 2 and queue 6, and queue 3 and queue 7. There is no restriction on EPD threshold per queue.
- The default chassis scheduler allocates resources for queue 0 through queue 3, with 25 percent of the bandwidth allocated to each queue. When you configure the chassis to use more than four queues, you must configure and apply a custom chassis scheduler to override the default. To apply a custom chassis scheduler, include the **scheduler-map-chassis** statement at the **[edit class-of-service interfaces at-*fpc/pic/***]** hierarchy level. For more information about configuring and applying a custom chassis scheduler, see "Applying Scheduler Maps to Chassis-Level Queues" on page 778.

Example: Enabling Eight Queues on ATM2 IQ Interfaces

In Figure 64 on page 843, Router A generates IP packets with different IP precedence settings. Router B is an M320 router or a T Series router with two ATM2 IQ interfaces. On Router B, interface **at-6/1/0** receives traffic from Router A, while interface **at-0/1/0** sends traffic to Router C. This example shows the CoS configuration for Router B.

Figure 64: Example Topology for Router with Eight Queues



On Router B:

```
[edit chassis]
fpc 0 {
   pic 1 {
      max-queues-per-interface 8;
   }
}
fpc 6 {
   pic 1 {
      max-queues-per-interface 8;
   }
}

[edit interfaces]
at-0/1/0 {
   atm-options {
      linear-red-profiles {
         red_1 queue-depth 1k high-plp-threshold 50 low-plp-threshold 80;
         red_2 queue-depth 2k high-plp-threshold 40 low-plp-threshold 70;
         red_3 queue-depth 3k high-plp-threshold 30 low-plp-threshold 60;
         red_4 queue-depth 4k high-plp-threshold 20 low-plp-threshold 50;
      }
      scheduler-maps {
         sch_red {
            vc-cos-mode strict;
            forwarding-class fc_q0 {
               priority high;
               transmit-weight percent 5;
               linear-red-profile red_1;
            }
            forwarding-class fc_q1 {
               priority low;
               transmit-weight percent 10;
               linear-red-profile red_2;
            }
            forwarding-class fc_q2 {
               priority low;
               transmit-weight percent 15;
               linear-red-profile red_3;
            }
            forwarding-class fc_q3 {
               priority low;
               transmit-weight percent 20;
               linear-red-profile red_4;
            }
            forwarding-class fc_q4 {
               priority low;
               transmit-weight percent 5;
               linear-red-profile red_1;
            }
            forwarding-class fc_q5 {
               priority low;
               transmit-weight percent 10;
               linear-red-profile red_2;
            }
            forwarding-class fc_q6 {
```


Copyright © 2019, Juniper Networks, Inc.

```
            priority low;
            transmit-weight percent 15;
            linear-red-profile red_3;
          }
          forwarding-class fc_q7 {
            priority low;
            transmit-weight percent 20;
            linear-red-profile red_4;
          }
        }
        sch_epd {
          vc-cos-mode alternate;
          forwarding-class fc_q0 {
            priority high;
            transmit-weight percent 5;
            epd-threshold 1024;
          }
          forwarding-class fc_q1 {
            priority low;
            transmit-weight percent 10;
            epd-threshold 2048;
          }
          forwarding-class fc_q2 {
            priority low;
            transmit-weight percent 15;
            epd-threshold 3072;
          }
          forwarding-class fc_q3 {
            priority low;
            transmit-weight percent 20;
            epd-threshold 4096;
          }
          forwarding-class fc_q4 {
            priority low;
            transmit-weight percent 5;
            epd-threshold 2048;
          }
          forwarding-class fc_q5 {
            priority low;
            transmit-weight percent 10;
            epd-threshold 3072;
          }
          forwarding-class fc_q6 {
            priority low;
            transmit-weight percent 15;
            epd-threshold 4096;
          }
          forwarding-class fc_q7 {
            priority low;
            transmit-weight percent 20;
            epd-threshold 5120;
          }
        }
      }
    }
```

Copyright © 2019, Juniper Networks, Inc.

```
    atm-options {
      vpi 0;
    }
    unit 0 {
      vci 0.100;
      shaping {
        cbr 1920000;
      }
      atm-scheduler-map sch_red;
      family inet {
        address 172.16.0.1/24;
      }
    }
    unit 1 {
      vci 0.101;
      shaping {
        vbr peak 1m sustained 384k burst 256;
      }
      atm-scheduler-map sch_epd;
      family inet {
        address 172.16.1.1/24;
      }
    }
}
at-6/1/0 {
    atm-options {
      vpi 0;
    }
    unit 0 {
      vci 0.100;
      family inet {
        address 10.10.0.1/24;
      }
    }
    unit 1 {
      vci 0.101;
      family inet {
        address 10.10.1.1/24;
      }
    }
}

[edit class-of-service]
classifiers {
    inet-precedence inet_classifier {
      forwarding-class fc_q0 {
        loss-priority low code-points 000;
      }
      forwarding-class fc_q1 {
        loss-priority low code-points 001;
      }
      forwarding-class fc_q2 {
        loss-priority low code-points 010;
      }
      forwarding-class fc_q3 {
```


Copyright © 2019, Juniper Networks, Inc.

```
        loss-priority low code-points 011;
      }
      forwarding-class fc_q4 {
        loss-priority low code-points 100;
      }
      forwarding-class fc_q5 {
        loss-priority low code-points 101;
      }
      forwarding-class fc_q6 {
        loss-priority low code-points 110;
      }
      forwarding-class fc_q7 {
        loss-priority low code-points 111;
      }
    }
    forwarding-classes {
      queue 0 fc_q0;
      queue 1 fc_q1;
      queue 2 fc_q2;
      queue 3 fc_q3;
      queue 4 fc_q4;
      queue 5 fc_q5;
      queue 6 fc_q6;
      queue 7 fc_q7;
    }
    interfaces {
      at-6/1/0 {
        unit * {
          classifiers {
            inet-precedence inet_classifier;
          }
        }
      }
    }
}
[edit routing-options]
static {
    route 10.10.20.2/32 {
      next-hop at-0/1/0.0;
      retain;
      no-readvertise;
    }
    route 10.10.1.2/32 {
      next-hop at-0/1/0.1;
      retain;
      no-readvertise;
    }
}
```

To see the results of this configuration, you can issue the following operational mode commands:

- **show interfaces at-0/1/0 extensive**
- **show interfaces queue at-0/1/0**

Copyright © 2019, Juniper Networks, Inc.

- **show class-of-service forwarding-class**

Related Documentation

- Applying Scheduler Maps to Chassis-Level Queues on page 778
- Configuring Up to 16 Custom Forwarding Classes on page 214

Copying the Packet Loss Priority to the CLP Bit on ATM Interfaces

For a provider-edge (PE) router with customer edge (CE)-facing, egress, ATM2 IQ interfaces configured with standard AAL5 encapsulation, you can enable the packet loss priority (PLP) setting to be copied into the cell loss priority (CLP) bit.

NOTE: This configuration setting is not applicable to Layer 2 circuit encapsulations because the control word captures and preserves CLP information. For more information about Layer 2 circuit encapsulations, see the *Junos OS Network Interfaces Library for Routing Devices*.

By default, at egress ATM2 IQ interfaces configured with standard AAL5 encapsulation, the PLP information is not copied to the CLP bit. This means the PLP information is not carried beyond the egress interface onto the CE router.

You can enable the PLP information to be copied into the CLP bit by including the **plp-to-clp** statement:

```
plp-to-clp;
```

You can include this statement at the following hierarchy levels:

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

Related Documentation

- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351

Configuring CoS for L2TP Tunnels on ATM Interfaces

The Layer 2 Tunneling Protocol (L2TP) is often used to carry traffic securely between an L2TP network server (LNS) to an L2TP access concentrator (LAC). CoS is supported for L2TP session traffic to a LAC on platforms configured as an LNS that include egress IQ2 PICs. Supported routers are:

- M7i and M10i routers
- M120 routers

Copyright © 2019, Juniper Networks, Inc.

To enable session-aware CoS on an L2TP interface, include the **per-session-scheduler** statement at the **[edit interfaces unit *logical-unit-number*]** hierarchy level.

```
[edit interfaces interface-name unit logical-unit-number]
per-session-scheduler;
```

You also must set the IQ2 PIC mode for session-aware traffic shaping and set the number of bytes to add to or subtract from the packet before ATM cells are created. To configure these options on the ingress side of the tunnel, include the **ingress-shaping-overhead** and **mode session-shaping** statements at the **[edit chassis fpc *slot-number* pic *pic-number* traffic-manager]** hierarchy level.

```
[edit chassis fpc slot-number pic pic-number]
traffic-manager {
   ingress-shaping-overhead number;
   mode session-shaping;
}
```

Various limitations apply to this feature:

- Only 991 shapers are supported on each IQ2 PIC.
- Sessions in excess of 991 cannot be shaped (but they can be policed).
- There is no support for PPP multilinks.
- The overall traffic rate cannot exceed the L2TP traffic rate, or else random drops result.
- There is no support for logical interface scheduling and shaping at the ingress because all schedulers are now reserved for L2TP.
- There is no support for physical interface rate shaping at the ingress.

You can provide policing support for sessions with more than the 991 shapers on each IQ2 PIC. Each session can have four or eight different classes of traffic (queues). Each class needs its own policer; for example, one for voice and one for data traffic. The policer is configured within a **simple-filter** statement and only **forwarding class** is supported in the **from** clause. Only one policer can be referenced in each simple filter.

The following example shows a policer within a simple filter applied to two assured forwarding classes:

```
[edit firewall]
policer P1 {
   if-exceeding {
      bandwidth-limit 400k;
         burst-size-limit 1500;
      }
   then discard;
}
family inet {
   simple-filter SF-1 {
      term T-1 {
         from {
```

Copyright © 2019, Juniper Networks, Inc.

```
            forwarding-class [ af11 af21 ];
        }
        then policer P1;
    }
  }
}
```

You can also set the number of bytes to add to or subtract from the packet at the egress of the tunnel. To configure these options on the egress side of the tunnel, include the **egress-shaping-overhead** and **mode session-shaping** statements at the **[edit chassis fpc *slot-number* pic *pic-number* traffic-manager]** hierarchy level.

```
[edit chassis fpc slot-number pic pic-number]
traffic-manager {
  egress-shaping-overhead number;
  mode session-shaping;
}
```

Related Documentation

- ingress-shaping-overhead on page 1170
- mode (Layer 2 Tunneling Protocol Shaping) on page 1211
- egress-shaping-overhead on page 1098

Configuring CoS for ATM2 IQ Virtual Circuit Tunnels

This example configures ATM2 IQ virtual circuit (VC) tunnel CoS components:

```
[edit interfaces]
at-1/2/0 {
  atm-options {
    vpi 0;
    linear-red-profiles red-profile-1 {
      queue-depth 35000 high-plp-threshold 75 low-plp-threshold 25;
    }
    scheduler-maps map-1 {
      vc-cos-mode strict;
      forwarding-class best-effort {
        priority low;
        transmit-weight percent 25;
        linear-red-profile red-profile-1;
      }
    }
  }
  unit 0 {
    vci 0.128;
    shaping {
      vbr peak 20m sustained 10m burst 20;
    }
    atm-scheduler-map map-1;
    family inet {
      address 192.168.0.100/32 {
```


Copyright © 2019, Juniper Networks, Inc.

```
            destination 192.168.0.101;
        }
    }
  }
}
```

Related Documentation

- *ATM2 IQ VC Tunnel CoS Components Overview*

Applying IEEE 802.1p BA Classifiers to Ethernet VPLS Over ATM

You can apply an IEEE 802.1p behavior aggregate (BA) classifier to VPLS in a bridged Ethernet over ATM environment using ATM (RFC 1483) encapsulation. This extracts the Layer 2 (frame level) IEEE 802.1p information from the cells arriving on the ATM interface. Note that the interface must be configured for the Ethernet VPLS service over ATM links.

This example applies the classifier **atm-ether-vpls-classifier** to an ATM interface using **ether-vpls-over-atm-llc** encapsulation. This is not a complete CoS configuration example.

```
[edit class-of-service interfaces]
at-1/2/3 {
  unit 0 {
    (...) # Other CoS features
    classifiers {
      ieee-802.1 atm-ether-vpls-classifier; # Classifier defined elsewhere
    }
  }
}

[edit]
interface at-1/2/3 {
  atm-options {
    vpi 0;
  }
  unit 0 {
    encapsulation ether-vpls-over-atm-llc; # Required encapsulation type
    vci 0.100;
    family vpls;
  }
}
```

You must configure a routing instance for the VPLS as well:

```
[edit routing-instances]
cos-test-1 {
  instance-type vpls; #This is required
  interface at-1/2/3;
  route-distinguisher 10.10.10.10:1;
  vrf-target target:11111:1;
  protocols {
    vpls {
      site-range 10;
```

Copyright © 2019, Juniper Networks, Inc.

```
        site cos-test-v1-site1 {
          site-identifier 1;
        }
      }
    }
  }
```

The Layer 2 VPN classification on an ATM interface is limited to the Layer 2 granularity, not to each separate VLAN/VPLS instance. In other words, all of the VLAN/VPLS packets arriving on an ATM virtual circuit are classified by a single IEEE 802.1p classifier. The individual flow of each VLAN cannot be identified at this level.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Configuring Behavior Aggregate Classifiers on page 51
- Default IEEE 802.1p Classifier on page 43

Example: Combining Layer 2 and Layer 3 Classification on the Same ATM Physical Interface

With the ATM II IQ PIC installed on the M320 router with the Enhanced Type 3 FPC or the M120 router, you can combine Layer 2 and Layer 3 classifiers on the same ATM physical interface. However, you must apply the classifiers to different logical interfaces (units). The Layer 3 interface can belong to a Layer 3 VPN or VPLS routing instance and the Layer 2 interface can belong to a VPLS routing instance. If the Layer 3 interface belongs to a VPLS routing instance, only IPv4 DSCP or Internet precedence classification is supported. When the ATM interface is part of a Layer 3 VPN, both IPv4 and IPv6 DSCP or Internet precedence classification is supported.

This example applies a Layer 3 DSCP classifier named **dscp-1** and a Layer 2 IEEE 802.1 classifier named **ieee-1** to ATM interface **at-4/1/1** units 0 and 1. The **inet-precedence** Layer 3 classification is also supported but is not used in this example.

```
[edit]
class-of-service {
  interfaces {
    at-4/1/1 {
      unit 0 {
        classifiers {
          dscp dscp_1;
        }
        unit 1 {
          classifiers {
            ieee-802.1 ieee;
          }
        }
      }
    }
```


Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36

Applying Scheduler Maps to ATM Interfaces

To define a scheduler map, you associate it with a forwarding class. Each class is associated with a specific queue, as follows:

- **best-effort**—Queue 0
- **expedited-forwarding**—Queue 1
- **assured-forwarding**—Queue 2
- **network-control**—Queue 3

NOTE: For M320 and T Series routers only, you can configure more than four forwarding classes and queues. For more information, see "Enabling Eight Queues on ATM Interfaces" on page 842.

When you configure an ATM scheduler map, the Junos OS creates these CoS queues for a VC. The Junos OS prefixes each packet delivered to the VC with the next-hop rewrite data associated with each queue.

To configure an ATM scheduler map, include the **scheduler-maps** statement at the **[edit interfaces at-*fpc/pic/port* atm-options]** hierarchy level:

```
edit interfaces at-fpc/pic/port atm-options]
scheduler-maps map-name {
  forwarding-class class-name {
    epd-threshold cells plp1 cells;
    linear-red-profile profile-name;
    priority (high | low);
    transmit-weight (cells number | percent number);
  }
  vc-cos-mode (alternate | strict);
}
```

You can define the following options for each forwarding class:

- **epd-threshold**—An EPD threshold provides a queue of cells that can be stored with tail drop. When a beginning-of-packet (BOP) cell is received, the VC's queue depth is checked against the EPD threshold. If the VC's queue depth exceeds the EPD threshold, the BOP cell and all subsequent cells in the packet are discarded.
- **linear-red-profile**—A linear RED profile defines the number of cells using the **queue-depth** statement within the RED profile. (You configure the **queue-depth** statement at the **[edit interfaces at-*fpc/pic/port* atm-options linear-red-profile *profile-name*]** hierarchy level.)

Copyright © 2019, Juniper Networks, Inc.

By default, if you include the **scheduler-maps** statement at the **[edit interfaces at-*fpc/pic/port* atm-options]** hierarchy level, the interface uses an EPD threshold that is determined by the Junos OS based on the available bandwidth and other parameters. You can override the default EPD threshold by setting an EPD threshold or a linear RED profile.

If shaping is enabled, the default EPD threshold is proportional to the shaping rate according to the following formula:

```
default epd-threshold = number of buffers * shaping rate / line rate
```

The minimum value is 48 cells. If the formula results in an EPD threshold less than 48 cells, the result is ignored, and the minimum value of 48 cells is used.

- priority—By default, queue 0 is high priority, and the remaining queues are low priority. You can configure high or low queuing priority for each queue.
- transmit-weight—By default, the transmit weight is 95 percent for queue 0, and 5 percent for queue 3. You can configure the transmission weight in number of cells or percentage. Each CoS queue is serviced in WRR mode. When CoS queues have data to send, they send the number of cells equal to their weight before passing control to the next active CoS queue. This allows proportional bandwidth sharing between multiple CoS queues within a rate-shaped VC tunnel. A CoS queue can send from 1 through 32,000 cells or from 5 through 100 percent of queued traffic before passing control to the next active CoS queue within a VC tunnel.

The AAL5 protocol prohibits cells from being interleaved on a VC; therefore, a complete packet is always sent. If a CoS queue sends more cells than its assigned weight because of the packet boundary, the deficit is carried over to the next time the queue is scheduled to transmit. If the queue is empty after the cells are sent, the deficit is waived, and the queue's assigned weight is reset.

NOTE: If you include the **scheduler-maps** statement at the **[edit interfaces at-*fpc/pic/port* atm-options]** hierarchy level, the **epd-threshold** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** or **[edit interfaces *interface-name* unit *logical-unit-number* address *address* family *family* multipoint-destination *address*]** hierarchy level has no effect because either the default EPD threshold, the EPD threshold setting in the forwarding class, or the linear RED profile takes effect instead.

Related Documentation

- Configuring Schedulers on page 258
- Configuring Scheduler Maps on page 258

Copyright © 2019, Juniper Networks, Inc.

Configuring ATM Scheduler Support for Ethernet VPLS over ATM Bridged Interfaces

You can configure ATM scheduler maps on Ethernet VPLS over bridged ATM interfaces.

Before you begin, you must have done the following tasks:

- Properly configured the router basics
- Verified you have support for VPLS and routing instance configuration
- Installed ATM II IQ PICs

When you configure ATM scheduler maps on Ethernet VPLS over bridged ATM interfaces, you can assign ATM traffic to various forwarding classes and queues. This feature is only available with the ATM II IQ PIC with Ethernet VPLS-over-ATM encapsulation.

The configuration takes place in four steps: define the scheduler map for ATM options on the interface, set the encapsulation type to Ethernet VPLS over ATM LLC, attach the scheduler map to the logical interface (unit), and include the interface in the VPLS routing instance configuration.

To configure ATM scheduler maps on Ethernet VPLS over bridged ATM interfaces:

1. Define the scheduler map for ATM options on the interface:

```
[edit interfaces at-fpc/pic/port]
user@host# set atm-options pic-type atm2
user@host# set atm-options vpi vpi-number
user@host# set atm-options scheduler-maps scheduler-map-name forwarding-class
   forwarding-class-name forwarding-class-option-statements)
(repeat last set command as necessary)
```

2. Set the encapsulation type to Ethernet VPLS over ATM LLC:

```
[edit interfaces at-fpc/pic/port]
user@host# set unit unit-number encapsulation ether-vpls-over-atm-llc
user@host# set unit unit-number vci vci-number
```

3. Attach the scheduler map to the logical interface (unit):

```
[edit interfaces at-fpc/pic/port]
user@host# set unit unit-number atm-scheduler-map scheduler-map-name
```

4. Include the interface in the VPLS routing instance configuration:

```
[edit interfaces at-fpc/pic/port]
user@host# top
[edit]
user@host# edit routing-instances routing-instance-name
[edit routing-instances routing-instance-name]
```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set interface at-fpc/pic/port.unit-number
user@host# set route-distinguisher value
user@host# set vrf-target target-value
user@host# set protocols vpls site-range value
user@host# set protocols vpls site site-name site-identifier number
```

When you are done, the configuration statements you added should look like the listings below.

1. The scheduler map for ATM options on the interface:

```
[edit interfaces at-fpc/pic/port atm-options]
pic-type atm2;
vpi vpi-number;
scheduler-maps {
  scheduler-map-name {
    forwarding-class forwarding-class-name {
      (forwarding-class option statements);
    }
  }
}
```

2. The encapsulation type to Ethernet VPLS over ATM LLC:

```
[edit interfaces at-fpc/pic/port unit unit-number]
encapsulation ether-vpls-over-atm-llc;
vci vci-number;
```

3. The scheduler map to the logical interface (unit):

```
[edit interfaces at-fpc/pic/port unit unit-number]
atm-scheduler-map scheduler-map-name;
```

4. The interface in the VPLS routing instance configuration:

```
[edit routing-instances routing-instance-name]
interface at-fpc/pic/port.unit-number;
route-distinguisher value;
vrf-target target-value;
protocols {
  vpls {
    site-range value;
    site site-name {
      site-identifier number;
    }
  }
}
```


Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Example: Configuring ATM Schedulers for Ethernet VPLS over ATM Bridged Interfaces on page 857

Example: Configuring ATM Schedulers for Ethernet VPLS over ATM Bridged Interfaces

The following example configures an ATM scheduler map named **cos-vpls** and attaches it to the ATM interface **at-1/0/0.0**, configures **ether-vpls-over-atm-llc** encapsulation, attaches the **cos-vpls** scheduler map to the logical interface (unit), and configures the ATM interface **at-1/0/0.0** as part of a VPLS routing instance named **cos-vpls-1**.

```
[edit]
interfaces {
   at-1/0/0 {
      atm-options {
         pic-type atm2;
         vpi 0;
         scheduler-maps {
            cos0 {
               forwarding-class assured-forwarding {
                  priority low;
                  transmit-weight percent 10;
               }
               forwarding-class best-effort {
                  priority low;
                  transmit-weight percent 20;
               }
               forwarding-class expedited-forwarding {
                  priority low;
                  transmit-weight percent 30;
               }
               forwarding-class network-control {
                  priority high;
                  transmit-weight percent 40;
               }
            }
         }
      }
      unit 0 {
         encapsulation ether-vpls-over-atm-llc;
         vci 0.1000;
         shaping {
            cbr 33k;
         }
         atm-scheduler-map cos0;
      }
   }
}

[edit]
routing-instances {
   cos-vpls-1 {
      instance-type vpls;
```

Copyright © 2019, Juniper Networks, Inc.

```
        interface at-1/0/0.0;
        route-distinguisher 10.255.245.51:1;
        vrf-target target:1234:1;
        protocols {
          vpls {
            site-range 10;
            no-tunnel-services;
            site vpls-1-site-1 {
              site-identifier 1;
            }
          }
        }
      }
    }
```

Related Documentation

- Configuring ATM Scheduler Support for Ethernet VPLS over ATM Bridged Interfaces on page 855
- Configuring Schedulers on page 258
- Configuring Scheduler Maps on page 258

Applying Scheduler Maps to Logical ATM Interfaces

To apply the ATM scheduler map to a logical interface, include the **atm-scheduler-map** statement:

```
atm-scheduler-map (map-name | default);
```

When you add or change a scheduler map, the associated logical interface is taken offline and then brought back online immediately. For ATM CoS to take effect, you must configure the VCI and VPI identifiers and traffic shaping on each VC by including the following statements:

```
vci vpi-identifier.vci-identifier;
shaping {
  (cbr rate | rtvbr peak rate sustained rate burst length  |vbr peak rate sustained rate burst
    length);
}
```

You can include these statements at the following hierarchy levels:

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

You can also apply a scheduler map to the chassis traffic that feeds the ATM interfaces. For more information, see "Applying Scheduler Maps to Chassis-Level Queues" on page 778.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation
- Configuring Scheduler Maps on page 258
- Applying Scheduler Maps Overview on page 259

Configuring Linear RED Profiles on ATM Interfaces

Linear random early detection (RED) profiles define CoS virtual circuit drop profiles. You can configure up to 32 linear RED profiles per port. 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.

To configure linear RED profiles, include the **linear-red-profiles** statement at the **[edit interfaces at-*fpc/pic/port* atm-options]** hierarchy level:

```
[edit interfaces at-fpc/pic/port atm-options]
linear-red-profiles profile-name {
  high-plp-max-threshold percent;
  low-plp-max-threshold percent;
  queue-depth cells high-plp-threshold percentlow-plp-threshold percent;
}
```

The **queue-depth**, **high-plp-threshold**, and **low-plp-threshold** statements are mandatory.

You can define the following options for each RED profile:

- **high-plp-max-threshold**—Define the drop profile fill-level for the high packet loss priority (PLP) CoS VC. When the fill level exceeds the defined percentage, all packets with high PLP are dropped.
- **low-plp-max-threshold**—Define the drop profile fill-level for the low PLP CoS VC. When the fill level exceeds the defined percentage, all packets with low PLP are dropped.
- **queue-depth**—Define maximum queue depth in the CoS VC drop profile. Packets are always dropped beyond the defined maximum. The range you can configure is from 1 through 64,000 cells.
- **high-plp-threshold**—Define CoS VC drop profile fill-level percentage when linear RED is applied to cells with high PLP. When the fill level exceeds the defined percentage, packets with high PLP are randomly dropped by RED.
- **low-plp-threshold**—Define CoS VC drop profile fill-level percentage when linear RED is applied to cells with low PLP. When the fill level exceeds the defined percentage, packets with low PLP are randomly dropped by RED.

Related Documentation
- *ATM2 IQ VC Tunnel CoS Components Overview*

Configuring Virtual Circuit CoS Mode on ATM Interfaces

Virtual Circuit (VC) CoS mode defines the CoS queue scheduling priority. By default, the VC CoS mode is **alternate**. When it is a queue's turn to transmit, the queue transmits up

Copyright © 2019, Juniper Networks, Inc.

to its weight in cells as specified by the **transmit-weight** statement at the **[edit interfaces at-*fpc/pic/port* atm-options scheduler-maps *map-name* forwarding-class *class-name*]** hierarchy level. The number of cells transmitted can be slightly over the configured or default transmit weight, because the transmission always ends at a packet boundary.

To configure the VC CoS mode, include the **vc-cos-mode** statement at the **[edit interfaces at-*fpc/pic/port* atm-options scheduler-maps]** hierarchy level:

```
edit interfaces at-fpc/pic/port atm-options scheduler-maps]
vc-cos-mode (alternate | strict);
```

Two modes of CoS scheduling priority are supported:

- **alternate**—Assign **high** priority to one queue. The scheduling of the queues alternates between the **high** priority queue and the remaining queues. Every other scheduled packet is from the **high** priority queue.
- **strict**—Assign strictly **high** priority to one queue. A queue with strictly **high** priority is always scheduled before the remaining queues. The remaining queues are scheduled in round-robin fashion.

Related Documentation

- *ATM2 IQ VC Tunnel CoS Components Overview*
- Priority Scheduling Overview on page 327

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 23

Configuring Class of Service on SONET/SDH OC48/STM16 IQE PICs

- CoS on SONET/SDH OC48/STM16 IQE PIC Overview on page 862
- Packet Classification on SONET/SDH OC48/STM16 IQE PICs on page 864
- Translation Table on SONET/SDH OC48/STM16 IQE PICs on page 865
- Configuring Translation Tables on SONET/SDH OC48/STM16 IQE PICs on page 866
- Example: Configuring CoS Value Translation Tables on SONET/SDH OC48/STM16 IQE PICs on page 867
- Scheduling and Shaping on SONET/SDH OC48/STM16 IQE PICs on page 870
- Configuring Scheduling, Shaping, and Priority Mapping on SONET/SDH OC48/STM16 IQE PICs on page 872
- Priority Mapping on SONET/SDH OC48/STM16 IQE PICs on page 873
- Example: Configuring Priority Scheduling on SONET/SDH OC48/STM16 IQE PICs on page 874
- Scaling for SONET/SDH OC48/STM16 IQE PICs on page 876
- Transmission Rate with Intelligent Oversubscription on SONET/SDH OC48/STM16 IQE PICs Overview on page 877
- Example: Configuring Transmit Rates That Add Up to More Than 100 Percent on page 881
- Example: Configuring Transmission Rate with Intelligent Oversubscription on SONET/SDH OC48/STM16 IQE PICs on page 882
- Example: Configuring a CIR and a PIR on SONET/SDH OC48/STM16 IQE Interfaces on page 890
- MDRR on SONET/SDH OC48/STM16 IQE PICs on page 891
- Configuring MDRR on SONET/SDH OC48/STM16 IQE PICs on page 891
- Example: Configuring MDRR on SONET/SDH OC48/STM16 IQE PICs on page 891
- WRED on SONET/SDH OC48/STM16 IQE PICs on page 891
- Configuring WRED on SONET/SDH OC48/STM16 IQE PICs on page 891
- Example: Configuring WRED on SONET/SDH OC48/STM16 IQE PICs on page 892
- Excess Bandwidth Sharing on SONET/SDH OC48/STM16 IQE PICs on page 892

Copyright © 2019, Juniper Networks, Inc.

- Configuring Excess Bandwidth Sharing on SONET/SDH OC48/STM16 IQE PICs on page 892
- Egress Rewrite on SONET/SDH OC48/STM16 IQE PICs on page 892
- Configuring Rewrite Rules on SONET/SDH OC48/STM16 IQE PIC on page 892
- Forwarding Class to Queue Mapping on SONET/SDH OC48/STM16 IQE PICs on page 893
- Configuring Forwarding Classes on SONET/SDH OC48/STM16 IQE PIC on page 893
- Configuring Rate Limits on SONET/SDH OC48/STM16 IQE PICs on page 894
- Example: Configuring Rate Limits on SONET/SDH OC48/STM16 IQE PICs on page 894

CoS on SONET/SDH OC48/STM16 IQE PIC Overview

The SONET/SDH OC48/STM16 IQE PIC is a clear-channel PIC that is designed to provide better scaling and improved queuing, buffering, and traffic shaping along with clear-channel functionality. Class of service (CoS) on the SONET/SDH OC48/STM16 IQE PIC supports per data-link connection identifier (DLCI) queuing at egress. The SONET/SDH OC48/STM16 IQE PIC can be used in Juniper Networks M320, MX240, MX960, T640, and T1600 routers.

The SONET/SDH OC48/STM16 IQE PIC supports the following CoS features:

- Eight queues per logical interface.

NOTE: Queue configuration in other modes, such as 4 queues per scheduler, is not supported on the SONET/SDH OC48/STM16 IQE PIC.

- Two shaping rates: a committed information rate (CIR) and peak information rate (PIR) per data-link connection identifier (DLCI).
- Sharing of excess bandwidth among logical interfaces.
- Five levels of priorities—three priorities for traffic below the guaranteed rate and two priorities for traffic above the guaranteed rate. By default, a strict-high queue gets the excess high priority and all other queues get the excess low priority.
- Ingress behavior aggregate (BA) classification.
- Translation table and egress rewrite.
- Egress delay buffer of 214ms.
- Forwarding class to queue remapping per DLCI.
- Weighted round-robin (WRR), weighted random early detection (WRED).
- Rate limit on all queues to limit the transmission rate.
- Per unit scheduling via DLCI at egress, where each DLCI gets a dedicated set of queues and a scheduler. When per unit scheduling is configured, the shaping can be configured at the logical and physical interface levels.

Copyright © 2019, Juniper Networks, Inc.

NOTE: Because the SONET/SDH OC48/STM16 IQE PIC is not an oversubscribed PIC, there is no ingress queuing. Therefore, ingress scheduling or shaping is not supported in SONET/SDH OC48/STM16 IQE PIC.

- Packet or byte statistics are separately collected for ingress and egress queues. The SONET/SDH OC48/STM16 IQE PIC provides the following statistics:
  - Ingress statistics:
    - Per logical interface transmit and drop bytes/packets statistics (based on Layer 3).
    - Per physical interface traffic bytes/packets statistics (based on Layer 2).
  - Egress statistics:
    - Per queue transmit and drop bytes/packets statistics (based on Layer 2).
    - Per queue per color drop bytes/packets statistics (based on Layer 2).
    - Per logical interface transmit and drop bytes/packets statistics (based on Layer 3).
    - Per physical interface traffic bytes/packets statistics (based on Layer 2).

To configure the features mentioned above, include the corresponding class-of-service (CoS) statements at the [**edit class-of-service**] hierarchy level. The CoS configuration statements supported on the SONET/SDH OC48/STM16 IQE PIC are the same as the CoS configuration statements supported on the IQ2E PIC except for the following unsupported statements.

Unsupported configuration statements at the **[edit chassis]** hierarchy level:

- **max-queues-per-interface**
- **no-concatenate**
- **q-pic-large-buffer**
- **red-buffer-occupancy**
- **ingress-shaping-overhead**
- **traffic manager mode**

Unsupported configuration statements at the **[edit class-of-service]** hierarchy level:

- **input-excess-bandwidth-share**
- **input-traffic-control-profile**
- **per-session-scheduler**
- **simple-filter**

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Egress Rewrite on SONET/SDH OC48/STM16 IQE PICs on page 892
- Scheduling and Shaping on SONET/SDH OC48/STM16 IQE PICs on page 870
- MDRR on SONET/SDH OC48/STM16 IQE PICs on page 891
- WRED on SONET/SDH OC48/STM16 IQE PICs on page 891
- Excess Bandwidth Sharing on SONET/SDH OC48/STM16 IQE PICs on page 892
- Packet Classification on SONET/SDH OC48/STM16 IQE PICs on page 864
- Translation Table on SONET/SDH OC48/STM16 IQE PICs on page 865

Packet Classification on SONET/SDH OC48/STM16 IQE PICs

Packet classification is used to partition the packets into different classes of traffic. You can use three methods to classify a packet:

- Behavior aggregate (BA) classification
- Fixed classification
- Multifield classification

The SONET/SDH OC48/STM16 IQE PIC supports BA classification and fixed classification. It does not support multifield classification. However, multifield classification can be done at the Packet Forwarding Engine level using firewall filters, which overrides the classification done at the PIC level.

The BA classifier maps a class-of-service (CoS) value to a forwarding class and loss priority. The forwarding class determines the output queue. The loss priority is used by schedulers in conjunction with the weighted random early detection (WRED) algorithm to control packet discard during periods of congestion.

The SONET/SDH OC48/STM16 IQE PICs support the following BA classifiers:

- DSCP IP or IP precedence
- DSCP IPv6
- MPLS (EXP)

The fixed classification matches the traffic on a logical interface level. The following example classifies all traffic on logical unit zero to the queue corresponding to assured forwarding.

```
[edit class-of-service interfaces so-0/1/2 unit 0]
forwarding-class af;
```

If the classifiers are not defined explicitly, then the default classifiers are applied as follows:

- All MPLS packets are classified using the MPLS (EXP) classifier. If there is no explicit MPLS (EXP) classifier, then the default MPLS (EXP) classifier is applied.

Copyright © 2019, Juniper Networks, Inc.

- All IPv4 packets are classified using the IP precedence or DSCP classifier. If there is no explicit IP precedence or DSCP classifier, then the default IP precedence classifier is applied.
- All IPv6 packets are classified using the DSCP IPv6 classifier. If there is no explicit DSCP IPv6 classifier, then the default DSCP IPv6 classifier is applied.

Related Documentation

- Egress Rewrite on SONET/SDH OC48/STM16 IQE PICs on page 892
- Translation Table on SONET/SDH OC48/STM16 IQE PICs on page 865

Translation Table on SONET/SDH OC48/STM16 IQE PICs

On the SONET/SDH OC48/STM16 IQE PIC, the behavior aggregate (BA) translation tables are included for every logical interface (unit) protocol family configured on the logical interface. The proper default translation table is active even if you do not include any explicit translation tables. You can display the current translation table values with the **show class-of-service translation-table** command.

On M320, MX series, T640, and T1600 routers with SONET/SDH OC48/STM16 IQE PICs, you can replace the type-of-service (ToS) or DSCP or MPLS (EXP) bit value on the incoming packet header on a logical interface with a user-defined value. The new value is used for all class-of-service processing and is applied before any other class-of-service or firewall treatment of the packet. On the SONET/SDH OC48/STM16 IQE PIC, the values configured with the **translation-table** statement determines the new ToS bit values.

The SONET/SDH OC48/STM16 IQE PIC supports four types of translation tables: IP precedence, IPv4 DSCP, IPv6 DSCP, and MPLS (EXP). You can configure a maximum of eight tables for each supported type. If a translation table is enabled for a particular type of traffic, then BA classification of the same type must be configured for that logical interface. That is, if you configure an IPv4 translation table, you must configure IPv4 BA classification on the same logical interface.

You can define many translation tables, as long as they have distinct names. You apply a translation table to a logical interface at the **[edit class-of-service interfaces]** hierarchy level. Translation tables always translate “like to like.” For example, a translation table applied to MPLS traffic can translate only from received EXP bit values to new EXP bit values. That is, translation tables cannot translate, for instance, from DSCP bits to INET precedence code points.

With translation table, the original fields in the received packet are overwritten with the new values configured in the translation table and the old values will be lost.

Related Documentation

- Configuring Translation Tables on SONET/SDH OC48/STM16 IQE PICs on page 866
- Example: Configuring CoS Value Translation Tables on SONET/SDH OC48/STM16 IQE PICs on page 867

Copyright © 2019, Juniper Networks, Inc.

Configuring Translation Tables on SONET/SDH OC48/STM16 IQE PICs

The SONET/SDH OC48/STM16 IQE PIC supports four types of translation tables: IP precedence, IPv4 DSCP, IPv6 DSCP, and MPLS (EXP). You can configure a maximum of eight tables for each supported type. If a translation table is enabled for a particular type of traffic, then behavior aggregate (BA) classification of the same type must be configured for that logical interface. That is, if you configure an IPv4 translation table, you must configure IPv4 BA classification on the same logical interface.

To configure ToS translation on the SONET/SDH OC48/STM16 IQE PIC:

1. Access the class-of-service hierarchy:

```
[edit]
user@host# edit class-of-service
```

2. Define the type of translation table:

```
[edit class-of-service]
translation-table {
  (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 | to-exp-from-exp |
    to-inet-precedence-from-inet-precedence) table-name {
    to-code-point value from-code-points (* | [ values ]);
  }
}
```

On the SONET/SDH OC48/STM16 IQE PIC, incoming ToS bit translation is subject to the following rules:

- Locally generated traffic is not subject to translation.
- The **to-dscp-from-dscp** translation table type is not supported if an Internet precedence classifier is configured.
- The **to-inet-precedence-from-inet-precedence** translation table type is not supported if a DSCP classifier is configured.
- The **to-dscp-from-dscp** and **to-inet-precedence-from-inet-precedence** translation table types cannot be configured on the same unit.
- The **to-dscp-from-dscp** and **to-inet-precedence-from-inet-precedence** translation table types are supported for IPv4 packets.
- Only the **to-dscp-ipv6-from-dscp-ipv6** translation table type is supported for IPv6 packets.
- Only the **to-exp-from-exp** translation table type is supported for MPLS packets.

  The **from-code-points** statement establishes the values to match on the incoming packets. The **default** option is used to match all values not explicitly listed, and, as a single entry in the translation table, to mark all incoming packets on an interface the same way. The **to-code-point** statement establishes the target values for the translation. If an incoming packet header ToS bit configuration is not covered by

Copyright © 2019, Juniper Networks, Inc.

the translation table list and a * option is not specified, the ToS bits in the incoming packet header are left unchanged.

NOTE: Translation tables are not supported if fixed classification is configured on the logical interface.

Related Documentation

- Translation Table on SONET/SDH OC48/STM16 IQE PICs on page 865
- Example: Configuring CoS Value Translation Tables on SONET/SDH OC48/STM16 IQE PICs on page 867

Example: Configuring CoS Value Translation Tables on SONET/SDH OC48/STM16 IQE PICs

The following example translates incoming DSCP values to the new values listed in the translation table. All incoming DSCP values other than **111111**, **111110**, **000111**, and **100111** are translated to **000111**.

```
[edit class-of-service]
translation-table {
  to-dscp-from-dscp dscp-trans-table {
    to-code-point 000000 from-code-points 111111;
    to-code-point 000001 from-code-points 111110;
    to-code-point 111000 from-code-points [ 000111 100111 ];
    to-code-point 000111 from-code-points *;
  }
}
```

You must apply the translation table to the logical interface input on the SONET/SDH OC48/STM16 IQE PIC:

```
[edit class-of-service interfaces so-1/0/0 unit 0]
translation-table to-dscp-from-dscp dscp-trans-table;
```

If you try to configure mutually exclusive translation tables on the same interface unit, you get a warning message when you display or commit the configuration:

```
so-0/1/1 {
    unit 0 {
        translation-table {
            ##
            ## Warning: to-dscp-from-dscp and
to-inet-precedence-from-inet-precedence not allowed on same unit
            ##
            to-inet-precedence-from-inet-precedence inet-trans-table;
            to-dscp-from-dscp dscp-trans-table;
        }
```

Copyright © 2019, Juniper Networks, Inc.

```
    }
}
```

You can issue the following operational mode commands to verify your configuration:

- show class-of-service translation-table
- show class-of-service interface *interface-name*

To verify that the correct values are configured, use the **show class-of-service translation-table** command. The **show class-of-service translation-table** command displays the code points of all translation tables configured. All values are displayed, not just those configured:

```
user@host> show class-of-service translation-table
Translation Table: dscp-trans-table, Translation table type: dscp-to-dscp, Index:
 6761
  From Code point    To Code Point
  000000                     000111
  000001                     000111
  000010                     000111
  000011                     000111
  000100                     000111
  000101                     000111
  000110                     000111
  000111                     111000
  001000                     000111
  001001                     000111
  001010                     000111
  001011                     000111
  001100                     000111
  001101                     000111
  001110                     000111
  001111                     000111
  010000                     000111
  010001                     000111
  010010                     000111
  010011                     000111
  010100                     000111
  010101                     000111
  010110                     000111
  010111                     000111
  011000                     000111
  011001                     000111
  011010                     000111
  011011                     000111
  011100                     000111
  011101                     000111
  011110                     000111
  011111                     000111
  100000                     000111
  100001                     000111
  100010                     000111
  100011                     000111
  100100                     000111
  100101                     000111
  100110                     000111
```


Copyright © 2019, Juniper Networks, Inc.

```
100111                    111000
101000                    000111
101001                    000111
101010                    000111
101011                    000111
101100                    000111
101101                    000111
101110                    000111
101111                    000111
110000                    000111
110001                    000111
110010                    000111
110011                    000111
110100                    000111
110101                    000111
110110                    000111
110111                    000111
111000                    000111
111001                    000111
111010                    000111
111011                    000111
111100                    000111
111101                    000111
111110                    000001
111111                    000000
```

To verify that the configured translation table is applied to the correct interface, use the **show class-of-service interface *interface-name*** command. The **show class-of-service interface *interface-name*** command displays the translation tables applied to the IQE interface:

```
user@host> show class-of-service interface so-2/3/0
Logical interface: so-2/3/0.0, Index: 68
  Object                  Name                    Type                    Index

  Rewrite                 exp-default             exp (mpls-any)             29

  Classifier              dscp-default            dscp                        7

  Classifier              exp-default             exp                        10

  Translation Table       exp-trans-table         EXP_TO_EXP              61925
```

ToS translation on the SONET/SDH OC48/STM16 IQE PIC is a form of behavior aggregate (BA) classification. The SONET/SDH OC48/STM16 IQE PIC does not support multifield classification of packets at the PIC level. For more information about multifield classification, see "Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields" on page 97.

Related Documentation

- Configuring Translation Tables on SONET/SDH OC48/STM16 IQE PICs on page 866

Copyright © 2019, Juniper Networks, Inc.

Scheduling and Shaping on SONET/SDH OC48/STM16 IQE PICs

The SONET/SDH OC48/STM16 IQE PIC supports the following scheduling and the shaping behavior:

- Per unit scheduling via data-link connection identifier (DLCI) at egress, where each DLCI gets a dedicated set of queues and a scheduler.

  NOTE: Because the SONET/SDH OC48/STM16 IQE PIC is not an oversubscribed PIC, there is no ingress queuing. therefore, the ingress scheduling or shaping is not supported in SONET/SDH OC48/STM16 IQE PIC.

- When a per unit scheduling is configured , the shaping can be configured at the logical and the physical interface levels.
- In both guaranteed and excess regions, the traffic on queues at the same priority is scheduled in weighted-round-robin (WRR) discipline and there is no shaping at queue-level.

On SONET/SDH OC48/STM16 IQE interfaces, you can configure a CIR (guaranteed rate) and a PIR (shaping rate) per data-link connection identifier (DLCI). The configured rates are gathered into a traffic control profile. If you configure a traffic control profile with a CIR (guaranteed rate) only, the PIR (shaping rate) is set to the physical interface (port) rate.

The computation of CIR and PIR on logical interfaces is shown in Table 117 on page 870. X and Y are values configured from the command-line interface.

Table 117: Computation of CIR and PIR on the Logical Interfaces

Port Mode	Junos OS CLI Configuration		SONET/SDH OC48/STM16 IQE PIC	
	Configured CIR	Configured PIR	Computed CIR	Computed PIR
Default (no CIR or PIR configured on logical interface)	Not configured	Not configured	Port speed	Port speed
Both CIR and PIR are configured on logical interface	X	Y	X	Y
CIR Mode (CIR is configured on at least one logical interface)	X	Not configured	X	Port speed
	Not configured	Y	50 Kbps	Y
	Not configured	Not configured	50 Kbps	Port speed

Copyright © 2019, Juniper Networks, Inc.

Table 117: Computation of CIR and PIR on the Logical Interfaces (continued)

Port Mode	Junos OS CLI Configuration		SONET/SDH OC48/STM16 IQE PIC	
	Configured CIR	Configured PIR	Computed CIR	Computed PIR
PIR Mode (PIR is configured on at least one logical interface)	Not configured	Y	Y	Y
	Not configured	Not configured	Remaining (port speed minus sum of PIRs of other logical interfaces) bandwidth is equally divided.	Port speed

The SONET/SDH OC48/STM16 IQE PIC supports rate limit on all queues. The computation of rate limit is shown in Table 118 on page 871.

Table 118: Computation of Rate for the Rate Limit Configured on the Queue with Transmit Rate Percentage

Scenario	Configured CIR Value for the Logical Interface or DLCI	Configured PIR Value for the Logical Interface or DLCI	SONET/SDH OC48/STM16 IQE PIC
1	No	No	Port value
2	Yes	No	CIR value
3	No	Yes	PIR value
4	Yes	Yes	CIR value

NOTE: When the queue transmission rates are oversubscribed, the rate-limit option configured on any of the queues uses the configured rate limit values, although the transmission rates are oversubscribed.

Transmit Rate Adding Up to More than 100 Percent

The SONET/SDH OC48/STM16 IQE PIC supports the maximum bandwidth optimization by overconfiguring the bandwidth up to 300 percent.

When the sum of transmission rates for all queues exceeds 100 percent, the interface is in an oversubscribed state. At the time of oversubscription, the queues are split into three priority groups. :

- Strict-High
- High, Medium-High, and Medium-Low
- Low

This computation is done after the internal mapping of the excess priority or the excess rate.

Copyright © 2019, Juniper Networks, Inc.

The sum of transmission rates for all queues in each of the priority groups is less than or equal to 100 percent, thereby allowing the SONET/SDH OC48/STM16 IQE PICs to support the maximum bandwidth optimization by overconfiguring the available bandwidth up to 300 percent.

NOTE:

- The **remainder** option is not supported on an oversubscribed SONET/SDH OC48/STM16 IQE PIC. When the sum of transmission rates for all queues exceeds 100 percent, and if one or more queues are configured with the **remainder** option, a syslog error message is generated and the configuration is ignored.
- When the sum of transmission rates of all queues in any of the priority groups exceeds 100 percent, the commit fails and an error message is displayed.

Related Documentation

- Example: Configuring a CIR and a PIR on SONET/SDH OC48/STM16 IQE Interfaces on page 890
- Example: Configuring Transmit Rates That Add Up to More Than 100 Percent on page 881

Configuring Scheduling, Shaping, and Priority Mapping on SONET/SDH OC48/STM16 IQE PICs

To configure shaping, scheduling, and priority mapping on the SONET/SDH OC48/STM16 IQE PIC, include the following statements at the **[edit class-of-service]** and **[edit interfaces]** hierarchy levels of the configuration:

```
[edit class-of-service]
traffic-control-profiles profile-name {
  guaranteed-rate (percent percentage | rate);
  scheduler-map map-name;
  shaping-rate (percent percentage | rate);
}
interfaces {
  interface-name {
    unit logical-unit-number {
      dlci dlci-identifier;
      output-traffic-control-profile profile-name ;
    }
  }
schedulers {
  scheduler-name {
    buffer-size (seconds | percent percentage | remainder | temporal microseconds);
    excess-priority value ;
    excess-rate percent percentage
    priority priority-level;
    transmit-rate (percent percentage | rate | remainder) < rate-limit>;
  }
```


Copyright © 2019, Juniper Networks, Inc.

```
}

[edit interfaces interface-name]
per-unit-scheduler;
```

NOTE:

- As indicated in the configuration, the **scheduler-map** and **shaping-rate** statements can be included at the **[edit class-of-service interfaces** *interface-name* **unit** *logical-unit-number***]** hierarchy level. However, we do not recommend this configuration. Include the **output-traffic-control-profile** statement instead.
- The **excess-rate** or the **excess-priority** statements are mapped for a specific configuration and are ignored otherwise. These two statements are enabled only for the configuration similarity with the other IQE PICs configuration statements.

Related Documentation

- Example: Configuring Priority Scheduling on SONET/SDH OC48/STM16 IQE PICs on page 874

Priority Mapping on SONET/SDH OC48/STM16 IQE PICs

The SONET/SDH OC48/STM16 IQE PIC supports three priorities for traffic below the guaranteed rate and two priorities for traffic above the guaranteed rate. The mapping between Junos OS priorities and the SONET/SDH OC48/STM16 IQE PIC hardware priorities below and above the guaranteed rate (CIR) is shown in Table 119 on page 873. By default, a strict-high queue gets the excess high priority and all other queues get the excess low priority.

Table 119: Junos OS Priorities Mapped to SONET/SDH OC48/STM16 IQE PIC Hardware Priorities

Junos OS Priority	SONET/SDH OC48/STM16 IQE PIC Hardware Priority Below Guaranteed Rate For Logical Interfaces	SONET/SDH OC48/STM16 IQE PIC Hardware Priority Above Guaranteed Rate For Logical Interfaces (Excess Priority)
Strict-high	High	High
High	High	Low
Medium-high	Medium	Low
Medium-low	Medium	Low
Low	Low	Low

The SONET/SDH OC48/STM16 IQE PIC internally maps the excess priority and the excess rate to achieve configuration parity with the other IQE PICs.

Copyright © 2019, Juniper Networks, Inc.

The queue-level mapping for the excess priority and the excess rate is shown in Table 120 on page 874.

Table 120: Queue-Level Mapping for Excess Priority and Excess Rate

Configured Values				Mapped Values			
Transmit Priority	Transmit Rate %	Excess Rate %	Excess Priority	Transmit Priority	Transmit Rate %	Excess Rate %	Excess Priority
Strict-high	X	Any value	Any value	No mapping	No mapping	Ignored	Ignored
	0	Any value	Any value	No mapping	No mapping	Ignored	Ignored
High or Medium or Low	X	Any value	Any value	No mapping	No mapping	Ignored	Ignored
	0	X	High	Medium	X	Ignored	Low
			Low	Low	X	Ignored	Low
		0	Any value	No mapping	No mapping	Ignored	Ignored

NOTE: The value X is the configured rate.

The SONET/SDH OC48 IQE PIC maps the excess rate and the excess priority based on the following conditions:

- If the transmit priority is not strict-high, the transmit rate is zero, the excess rate is nonzero, and the excess priority is high, then the value of the transmit priority is changed to medium and the value of the excess priority is changed to low.
- If the transmit priority is not strict-high, the transmit rate is zero, the excess rate is nonzero, and the excess priority is low, then the value of the transmit priority is changed to low and the value of the excess priority is changed to low.
- In all the cases other than those mentioned above, the SONET/SDH OC48 IQE PIC ignores the excess rate and the excess priority configurations and generates the system log messages.

Related Documentation

- Example: Configuring Priority Scheduling on SONET/SDH OC48/STM16 IQE PICs on page 874

Example: Configuring Priority Scheduling on SONET/SDH OC48/STM16 IQE PICs

In the following example, **ef** is an expedited forwarding traffic queue; **af_01**, **af_02**, **af_03**, and**af_04** are assured forwarding traffic queues; **be** is a best effort forwarding queue; and **nc** is a network control traffic queue.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service]
traffic-control profiles tcp {
   shaping-rate 300M;
}
[edit class-of-service]
interfaces {
   so-2/2/0 {
      unit 0 {
         output-traffic-control-profiles tcp;
      }
   }
   schedulers {
   ef {
      transmit-rate percent 50 rate-limit;
      buffer-size percent 5;
      priority strict-high;
   }
   nc {
      transmit-rate percent 0;
      excess-rate percent 5;
      buffer-size percent 5;
      priority low;
      excess-priority high;
   }
   af_01 {
      transmit-rate percent 0;
      excess-rate percent 20;
      buffer-size percent 18;
      priority low;
      excess-priority low;
   }
   af_02 {
      transmit-rate percent 0;
      excess-rate percent 35;
      buffer-size percent 18;
      priority low;
      excess-priority low;
   }
   af_03 {
      transmit-rate percent 0;
      excess-rate percent 30;
      buffer-size percent 18;
      priority low;
      excess-priority low;
   }
   af_04 {
      transmit-rate percent 0;
      excess-rate percent 9;
      buffer-size percent 18;
      priority low;
      excess-priority low;
   }
   be {
      transmit-rate percent 0;
      excess-rate percent 1;
```

Copyright © 2019, Juniper Networks, Inc.

```
        buffer-size percent 18;
        priority low;
        excess-priority low;
    }
}
```

Table 121: Priority Mapping and Output Calculation for Different Queues on the SONET/SDH OC48/STM16 IQE PIC

Queue	Priority	Transmit Rate	Excess Priority on the SONET/SDH OC48/STM16 IQE PIC (Mapped to Guaranteed Priority)	Excess Rate on the SONET/SDH OC48/STM16 IQE PIC (Mapped to Transmit Rate)	Input (Mbps)	Output (Mbps)
ef	Strict-high	50 (50% of PIR=150 Mbps)	Not applicable	Not applicable	300	150
nc	Low	0	Excess high	5	300	150
af_01	Low	0	Excess low	20	300	0
af_02	Low	0	Excess low	35	300	0
af_03	Low	0	Excess low	30	300	0
af_04	Low	0	Excess low	9	300	0
be	Low	0	Excess low	1	300	0

As shown in Table 121 on page 876, the **ef** queue takes precedence over all queues and consumes 150 Mbps (50 percent of the PIR; that is, half of 300 Mpbs) bandwidth. The remaining 150 Mbps is rate limited. The **af_01**, **af_02**, **af_03**, **af_04** and the **be** queues do not get any bandwidth.

Because the rate limit is not configured on the **nc** queue, and it has the excess high priority, the **nc** queue consumes the remaining bandwidth of 150 Mbps.

Related Documentation

- CoS on SONET/SDH OC48/STM16 IQE PIC Overview on page 862
- Configuring MDRR on Enhanced Queuing DPCs on page 913
- Priority Scheduling Overview on page 327
- Configuring Schedulers for Priority Scheduling on page 330

Scaling for SONET/SDH OC48/STM16 IQE PICs

The scaling parameters for the SONET/SDH OC48/STM16 IQE PIC are defined in the Table 122 on page 877

Copyright © 2019, Juniper Networks, Inc.

Table 122: Scaling for SONET/SDH OC48/STM16 IQE PIC

Scaling Parameter on SONET/SDH OC48/STM16 IQE PIC	Value
Number of physical interfaces per PIC	4
Maximum queues per physical interface	8176
Maximum queues per PIC	16000
Maximum logical interface (DLCI) per PIC without per-unit scheduling	4083
Maximum logical interface (DLCI) per PIC with per-unit scheduling	2000

Related Documentation

- Configuring Scheduling, Shaping, and Priority Mapping on SONET/SDH OC48/STM16 IQE PICs on page 872
- Example: Configuring Rate Limits on SONET/SDH OC48/STM16 IQE PICs on page 894
- CoS on SONET/SDH OC48/STM16 IQE PIC Overview on page 862
- Configuring Rewrite Rules on SONET/SDH OC48/STM16 IQE PIC on page 892

Transmission Rate with Intelligent Oversubscription on SONET/SDH OC48/STM16 IQE PICs Overview

Junos OS class of service (CoS) enables you to treat traffic differently by providing a minimum bandwidth guarantee, low latency, low packet loss, or a combination of these properties for categories of traffic, called forwarding classes. When traffic reaches an outbound interface, traffic is queued for transmission on the physical media. The forwarding class determines the queuing of traffic and other functions of processing class of service, such as rewriting behavior aggregate markers.

You can control the way the system services queues by configuring schedulers and scheduler maps. After traffic is placed in the appropriate queues, a scheduler defines how an interface should process this traffic from each queue. A scheduler is associated with a particular queue and a forwarding class through a scheduler map.

The parameters in a scheduler that define how to service a queue include transmission rate, transmission priority, buffer size, and a random early detection (RED) algorithm. You can define the order in which packets transmit a queue by configuring a priority and transmission rate for each queue. The buffer size and RED configuration define the storage and dropping of packets for each queue.

Junos OS supports multiple levels of transmission priority, with higher-priority queues being serviced before lower-priority queues, as long as the higher-priority forwarding classes retain enough bandwidth credit. The priority levels are Strict-High, High, Medium-High, Medium-Low, and Low. The priority scheduling of forwarding classes determines the order in which an outbound interface transmits traffic from the queues.

Copyright © 2019, Juniper Networks, Inc.

The transmission rate, on the other hand, controls how much bandwidth the traffic associated with a given forwarding class can consume. By default, all queues can exceed their assigned transmission rate if other queues are not fully utilizing their assigned rates, unless you configure the transmission rate with the **exact** option.

The transmission rate can be a fixed value, such as 1 megabit per second (Mbps), a percentage of the total available bandwidth, or the rest of the available bandwidth. You can limit the transmission bandwidth to the exact value you configure, or allow it to exceed the configured rate if additional bandwidth is available from other queues. This property enables you to ensure that each queue receives the amount of bandwidth appropriate to its level of service.

The SONET/SDH OC48/STM16 IQE PIC is a clear-channel PIC that is designed to provide better scaling and improved queuing, buffering, and traffic shaping along with clear-channel functionality. The PIC is preconfigured with five levels of priorities with three priorities for traffic below the guaranteed rate (CIR) and two priorities for traffic above the guaranteed rate (PIR).

Oversubscription is a state where the transmission rate of the incoming packet is much higher than the rate the Packet Forwarding Engine and system can handle, causing important packets to be dropped. If an oversubscribed link or service experiences an excess of traffic—either bursty or non-bursty—it can result in traffic loss or delay that could potentially affect other services and links. To reduce the risks of oversubscription, the SONET/SDH OC48/STM16 IQE PIC ensures prioritization and allows mission-critical services to be protected during congestion with intelligent dropping of packets. In addition to dropping low priority packets during congestion, the PIC prevents quality deterioration in periods of high traffic with intelligent sharing of excess bandwidth, providing bandwidth optimization.

Previously, the SONET/SDH OC48/STM16 IQE PIC supported a maximum bandwidth optimization by oversubscribing the available bandwidth up to 200 percent. This optimization was achieved by excluding the transmission rate percentage specified for the Strict-High queue from the total 100 percent transmission rate. Therefore, the transmission rate percentage for all the non-Strict-High queues added up to 100 percent. This computation was done after the internal mapping of the excess priority or the excess rate.

At the time of oversubscription, the queues were split into two priority groups:

- Strict-High
- High, Medium-High, Medium-Low, and Low

As an enhancement to the intelligent oversubscription feature on SONET/SDH OC48/STM16 IQE PICs, support for maximum bandwidth optimization is increased to 300 percent with an additional priority group being created for all queues marked with low priority.

Copyright © 2019, Juniper Networks, Inc.

When the sum of transmission rates for all queues exceeds 100 percent, the interface is in an oversubscribed state. At the time of oversubscription, the queues are split into three priority groups with the intelligent oversubscription feature enhancement:

- Strict-High
- High, Medium-High, and Medium-Low
- Low

The sum of transmission rates for all queues in each of the priority groups is less than or equal to 100 percent, thereby allowing the SONET/SDH OC48/STM16 IQE PICs to support the maximum bandwidth optimization by overconfiguring the available bandwidth up to 300 percent.

NOTE: When the sum of transmission rates of all queues in any of the priority groups exceeds 100 percent, the commit fails and an error message is displayed.

When the sum of transmission rates for all queues exceeds 100 percent, the configured transmission rates are scaled down to 100 percent. This is called rebasing and is required for accurate mapping of transmission rates to the weights assigned to each queue. Weights are assigned to a queue based on the queue properties and are used to determine the distribution of available bandwidth and flow of traffic from each queue.

Configuring any of the queues with the **remainder** option on an oversubscribed SONET/SDH OC48/STM16 IQE PIC is not allowed. When a queue is configured with the **remainder** option, and the sum of transmission rates for all non-remainder queues is less than or equal to 100 percent, rebasing is not required. However, calculating the remainder transmission rate for a queue configured with the **remainder** option differs.

NOTE: The remainder option is not supported on an oversubscribed SONET/SDH OC48/STM16 IQE PIC. When the sum of transmission rates for all queues exceeds 100 percent, and if one or more queues are configured with the remainder option, a syslog error message is generated and the configuration is ignored.

Previously, the remainder calculation on a SONET/SDH OC48/STM16 IQE PIC did not include the queues specified for Strict-High priority. The remainder transmission rate was calculated by taking into consideration transmission rates for all other queues excluding the sum of transmission rates of all Strict-High queues. With the enhancement to the intelligent oversubscription on a SONET/SDH OC48/STM16 IQE PIC, the remainder transmission rate is calculated by taking into consideration the transmission rates of all other queues irrespective of the priority specified.

The rebased values are only used for assigning weights to queues, which affect the order in which the queues are serviced. If any queue that qualifies for rebasing is configured

Copyright © 2019, Juniper Networks, Inc.

with the rate-limit option, weights are assigned to queues after applying the configured value of rate-limit for that particular queue.

As an example, sample configurations A and B in Table 123 on page 880 display the need for rebasing transmission rates and **remainder** calculation.

Table 123: Rebasing Transmission Rates and Remainder Calculation

Queue	Priority	Transmission rate
Configuration A		
q0	Strict-High	100%
q1	High	30%
q2	Medium-High	30%
q3	Medium-Low	30%
q4	Low	20%
q5	Low	20%
q6	Low	20%
q7	Low	remainder
Configuration B		
q0	Strict-High	30%
q1	Medium	20%
q2	Low	40%
q3	Low	remainder

In Configuration A, the sum of transmission rates of non-remainder queues (q0, q1, q2, q3, q4, q5, and q6) exceeds 100 percent, leaving the interface in an oversubscribed state. Because configuring the **remainder** option is not supported on an oversubscribed PIC, and q7 is a remainder queue, Configuration A is ignored, although all the queues qualify for rebasing.

However, if the sum of transmission rates for all the queues exceeded 300 percent, or if the sum of transmission rates for all queues in any of the priority groups exceeded 100 percent, the configuration is ignored.

In Configuration B, the sum of transmission rates of non-remainder queues (q0, q1, and q2) is less than 100 percent. Therefore, rebasing of transmission rates is not required. Remainder calculation for q3 is done by deducting the sum of transmission rates of

Copyright © 2019, Juniper Networks, Inc.

non-remainder queues from 100, irrespective of the priority specified. In this example, the transmission rate for q3 is (100 - 30 - 20 - 40) 10%.

The support for oversubscribing the bandwidth on a SONET/SDH OC48/STM16 IQE PIC up to 300 percent increases the efficiency of networks and reduces CapEx for network operators. Large service providers have exacting performance requirements, and the impact of traffic disruptions due to congestion on an oversubscribed interface can be significant. The SONET/SDH OC48/STM16 IQE PIC virtually eliminates this risk with intelligent oversubscription capabilities that enable carriers to ensure the performance of mission-critical services on oversubscribed interfaces and routers.

Related Documentation

- Example: Configuring Transmission Rate with Intelligent Oversubscription on SONET/SDH OC48/STM16 IQE PICs on page 882

Example: Configuring Transmit Rates That Add Up to More Than 100 Percent

In the following example, **ef** is an expedited forwarding traffic queue; **nc** is a network control traffic queue; **af_01**, **af_02**, **af_03**, and **af_04** are assured forwarding traffic queues; and **be** is a best effort forwarding queue. **so-2/2/0 unit 0** is the logical interface.

```
[edit class-of-service]
  traffic-control profiles tcp {
    shaping-rate 300M;
  }
  interfaces {
    so-2/2/0 {
      unit 0 {
        output-traffic-control-profiles tcp;
      }
    }
    schedulers {
    ef {
      transmit-rate percent 50 rate-limit;
      priority strict-high;
    }
    nc {
      transmit-rate percent 5;
      priority high;
    }
    af_04 {
      transmit-rate percent 20;
      priority medium;
    }
    af_03 {
      transmit-rate percent 35;
      priority low;
    }
    af_02 {
      transmit-rate percent 30;
      priority low;
    }
```

Copyright © 2019, Juniper Networks, Inc.

```
    af_01 {
      transmit-rate percent 9;
      priority low;
    }
    be {
      transmit-rate percent 1;
      priority low;
    }
  }
```

The **ef** and the **nc** queues are at the same priority. Therefore, both these queues take precedence over all the other queues. The **ef** queue consumes 100 Mbps (50 percent of the CIR; that is, 50 percent of 200 Mpbs) bandwidth. The remaining 200 Mbps is rate limited. The **nc** queue continues to consume the bandwidth till the logical interface reaches its CIR of 200 Mbps. Therefore, the **nc** queue gets 100 Mbps bandwidth. When the logical interface reaches its CIR, all queues transition into the excess region and the scheduler allocates the remaining bandwidth to the non-expedited forwarding queues based on their default excess priorities and default excess rates (same as the transmit rates).

As per Table 127 on page 914, all the non-strict-high queues are in the same excess priority (in this case, low priority), these non-strict-high queues get the bandwidth out of the remaining 100 Mbps in the ratio of 5:20:35:30:9:1 until the logical interface consumes its shaping rate of 300 Mbps. Thus, the non-strict-high queues add up to 100 percent of bandwidth utilization to optimize the bandwidth usage.

Related Documentation

- CoS on SONET/SDH OC48/STM16 IQE PIC Overview on page 862
- Scheduling and Shaping on SONET/SDH OC48/STM16 IQE PICs on page 870

Example: Configuring Transmission Rate with Intelligent Oversubscription on SONET/SDH OC48/STM16 IQE PICs

This example shows how to configure transmission rates on a sonet interface for eight forwarding classes with transmission rate values that exceed 100 percent, causing the interface to be in an oversubscribed state.

- Requirements on page 882
- Overview and Topology on page 883
- Configuration on page 884
- Verification on page 889

Requirements

This example requires the following hardware and software components:

- Networking devices using a SONET/SDH OC48/STM16 IQE PIC.
- Junos OS Release 12.2 or later running on the devices.

Copyright © 2019, Juniper Networks, Inc.

Before you begin:

1. Configure the device interfaces.
2. Enable class-of-service (CoS) queuing, scheduling, and shaping on the device interfaces.

Overview and Topology

Junos OS Release 12.2 and later support oversubscribing the available bandwidth on a SONET/SDH OC48/STM16 IQE PIC up to 300 percent. This optimization is achieved by creating an additional priority group for all queues specified for low priority, and the sum of transmission rates for all the low priority queues adding up to 100 percent, independent of the transmission rate configured for all other queues.

Previously, the SONET/SDH OC48/STM16 IQE PIC supported a maximum bandwidth optimization by oversubscribing the available bandwidth up to 200 percent, by excluding the transmission rate percentage specified for the Strict-High queue from the total 100 percent transmission rate. Therefore, the transmission rate percentage for all the non-Strict-High queues added up to 100 percent. This computation was done after the internal mapping of the excess priority or the excess rate.

As an enhancement to the intelligent oversubscription feature on SONET/SDH OC48/STM16 IQE PICs, support for maximum bandwidth optimization is increased to 300 percent.

When the sum of transmission rates for all queues exceeds 100 percent, the interface is in an oversubscribed state. At the time of oversubscription, the queues are split into three priority groups with the intelligent oversubscription feature enhancement:

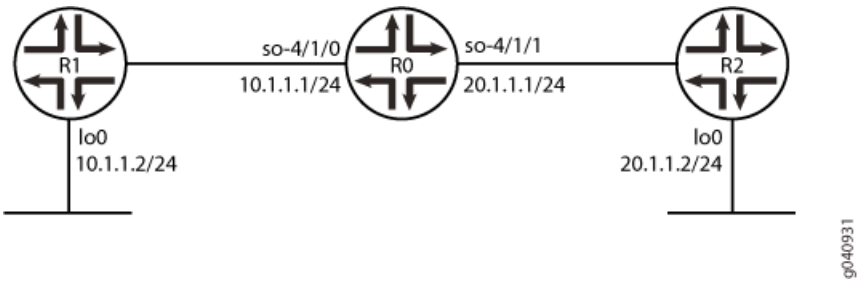
- Strict-High
- High, Medium-High, and Medium-Low
- Low

Each of the above priority groups can be configured to have a transmission rate oversubscription up to 100 percent. The transmission rate oversubscription value can be expressed as a percentage of the CIR or PIR value or as an absolute value.

NOTE: The remainder option is not supported on an oversubscribed SONET/SDH OC48/STM16 IQE PIC. When the sum of the transmission rates for all queues exceeds 100 percent, and if one or more queues are configured with the remainder option, a syslog error message is generated and the configuration is ignored.

Copyright © 2019, Juniper Networks, Inc.

In this example, Router R0 is the route on which the CoS options are configured. Routers R1 and R2 are directly connected to R0 and send traffic to R0.



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.

R0

```
set class-of-service classifiers inet-precedence inet_classy forwarding-class fc0
  loss-priority low code-points 000
set class-of-service classifiers inet-precedence inet_classy forwarding-class fc1
  loss-priority low code-points 001
set class-of-service classifiers inet-precedence inet_classy forwarding-class fc2
  loss-priority low code-points 010
set class-of-service classifiers inet-precedence inet_classy forwarding-class fc3
  loss-priority low code-points 011
set class-of-service classifiers inet-precedence inet_classy forwarding-class fc4
  loss-priority low code-points 100
set class-of-service classifiers inet-precedence inet_classy forwarding-class fc5
  loss-priority low code-points 101
set class-of-service classifiers inet-precedence inet_classy forwarding-class fc6
  loss-priority low code-points 110
set class-of-service classifiers inet-precedence inet_classy forwarding-class fc7
  loss-priority low code-points 111
set class-of-service forwarding-classes class fc0 queue-num 0
set class-of-service forwarding-classes class fc1 queue-num 1
set class-of-service forwarding-classes class fc2 queue-num 2
set class-of-service forwarding-classes class fc3 queue-num 3
set class-of-service forwarding-classes class fc4 queue-num 4
set class-of-service forwarding-classes class fc5 queue-num 5
set class-of-service forwarding-classes class fc6 queue-num 6
set class-of-service forwarding-classes class fc7 queue-num 7
set class-of-service traffic-control-profiles TCP scheduler-map map_ifls
set class-of-service traffic-control-profiles TCP shaping-rate 1g
set class-of-service interfaces so-4/1/0 unit 0 classifiers inet-precedence inet_classy
set class-of-service interfaces so-4/1/1 unit 0 output-traffic-control-profile TCP
set class-of-service schedulers s0 transmit-rate percent 25
set class-of-service schedulers s0 priority strict-high
set class-of-service schedulers s1 transmit-rate percent 20
set class-of-service schedulers s1 priority high
set class-of-service schedulers s2 transmit-rate percent 15
```


Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service schedulers s2 priority high
set class-of-service schedulers s3 transmit-rate percent 35
set class-of-service schedulers s3 priority medium-high
set class-of-service schedulers s4 transmit-rate percent 10
set class-of-service schedulers s4 priority medium-low
set class-of-service schedulers s5 transmit-rate percent 15
set class-of-service schedulers s5 priority low
set class-of-service schedulers s6 transmit-rate percent 15
set class-of-service schedulers s6 priority low
set class-of-service schedulers s7 transmit-rate percent 15
set class-of-service schedulers s7 priority low
set class-of-service scheduler-maps map_ifls forwarding-class fc0 scheduler s0
set class-of-service scheduler-maps map_ifls forwarding-class fc1 scheduler s1
set class-of-service scheduler-maps map_ifls forwarding-class fc2 scheduler s2
set class-of-service scheduler-maps map_ifls forwarding-class fc3 scheduler s3
set class-of-service scheduler-maps map_ifls forwarding-class fc4 scheduler s4
set class-of-service scheduler-maps map_ifls forwarding-class fc5 scheduler s5
set class-of-service scheduler-maps map_ifls forwarding-class fc6 scheduler s6
set class-of-service scheduler-maps map_ifls forwarding-class fc7 scheduler s7
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode*. To configure the R0 router:

1. Configure an IP Precedence classifier to classify incoming packets based on the code point values.

```
[edit class-of-service classifiers]
user@R0# set inet-precedence inet_classy
```

2. Define the classification of code point values to a forwarding class, and configure code point values to classify to loss priority Low.

```
[edit class-of-service classifiers inet-precedence inet_classy]
user@R0# set fc0 loss-priority low code-points 000
user@R0# set fc1 loss-priority low code-points 001
user@R0# set fc2 loss-priority low code-points 010
user@R0# set fc3 loss-priority low code-points 011
user@R0# set fc4 loss-priority low code-points 100
user@R0# set fc5 loss-priority low code-points 101
user@R0# set fc6 loss-priority low code-points 110
user@R0# set fc7 loss-priority low code-points 111
```

3. Define mapping of forwarding classes to queue numbers.

```
[edit class-of-service forwarding-classes class]
user@R0# set fc0 queue-num 0
user@R0# set fc1 queue-num 1
user@R0# set fc2 queue-num 2
user@R0# set fc3 queue-num 3
user@R0# set fc4 queue-num 4
```

Copyright © 2019, Juniper Networks, Inc.

```
user@R0# set fc5 queue-num 5
user@R0# set fc6 queue-num 6
user@R0# set fc7 queue-num 7
```

4. Configure traffic shaping and scheduling profiles.

```
[edit class-of-service traffic-control-profiles]
user@R0# set TCP scheduler-map map_ifls
user@R0# set TCP shaping-rate 1g
```

5. Apply the class-of-service options to interfaces.

```
[edit class-of-service interfaces]
user@R0# set so-4/1/0 unit 0 classifiers inet-precedence inet_classy
user@R0# set so-4/1/1 unit 0 output-traffic-control-profile TCP
```

6. Configure eight packet schedulers with scheduling priority and transmission rates.

```
[edit class-of-service schedulers]
user@R0# set s0 transmit-rate percent 25
user@R0# set s0 priority strict-high
user@R0# set s1 transmit-rate percent 20
user@R0# set s1 priority high
user@R0# set s2 transmit-rate percent 15
user@R0# set s2 priority high
user@R0# set s3 transmit-rate percent 35
user@R0# set s3 priority medium-high
user@R0# set s4 transmit-rate percent 0
user@R0# set s4 priority medium-low
user@R0# set s5 transmit-rate percent 15
user@R0# set s5 priority low
user@R0# set s6 transmit-rate percent 15
user@R0# set s6 priority low
user@R0# set s7 transmit-rate percent 15
user@R0# set s7 priority low
```

7. Define mapping of forwarding classes to packet schedulers.

```
[edit class-of-service scheduler-maps]
user@R0# set map_ifls forwarding-class fc0 scheduler s0
user@R0# set map_ifls forwarding-class fc1 scheduler s1
user@R0# set map_ifls forwarding-class fc2 scheduler s2
user@R0# set map_ifls forwarding-class fc3 scheduler s3
user@R0# set map_ifls forwarding-class fc4 scheduler s4
user@R0# set map_ifls forwarding-class fc5 scheduler s5
user@R0# set map_ifls forwarding-class fc6 scheduler s6
user@R0# set map_ifls forwarding-class fc7 scheduler s7
```


Copyright © 2019, Juniper Networks, Inc.

Results

From configuration mode, confirm your configuration by entering 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.

```
class-of-service {
  classifiers {
    inet-precedence inet_classy {
      forwarding-class fc0 {
        loss-priority low code-points 000;
      }
      forwarding-class fc1 {
        loss-priority low code-points 001;
      }
      forwarding-class fc2 {
        loss-priority low code-points 010;
      }
      forwarding-class fc3 {
        loss-priority low code-points 011;
      }
      forwarding-class fc4 {
        loss-priority low code-points 100;
      }
      forwarding-class fc5 {
        loss-priority low code-points 101;
      }
      forwarding-class fc6 {
        loss-priority low code-points 110;
      }
      forwarding-class fc7 {
        loss-priority low code-points 111;
      }
    }
  }
  forwarding-classes {
    class fc0 queue-num 0;
    class fc1 queue-num 1;
    class fc2 queue-num 2;
    class fc3 queue-num 3;
    class fc4 queue-num 4;
    class fc5 queue-num 5;
    class fc6 queue-num 6;
    class fc7 queue-num 7;
  }
  traffic-control-profiles {
    TCP {
      scheduler-map map_ifls;
      shaping-rate 1g;
    }
  }
  interfaces {
    so-4/1/0 {
```

Copyright © 2019, Juniper Networks, Inc.

```
      unit 0 {
        classifiers {
          inet-precedence inet_classy;
        }
      }
    }
    so-4/1/1 {
      unit 0 {
        output-traffic-control-profile TCP;
      }
    }
  }
  schedulers {
    s0 {
      transmit-rate percent 25;
      priority strict-high;
    }
    s1 {
      transmit-rate percent 20;
      priority high;
    }
    s2 {
      transmit-rate percent 15;
      priority high;
    }
    s3 {
      transmit-rate percent 35;
      priority medium-high;
    }
    s4 {
      transmit-rate percent 10;
      priority medium-low;
    }
    s5 {
      transmit-rate percent 15;
      priority low;
    }
    s6 {
      transmit-rate percent 15;
      priority low;
    }
    s7 {
      transmit-rate percent 15;
      priority low;
    }
  }
  scheduler-maps {
    map_ifls {
      forwarding-class fc0 scheduler s0;
      forwarding-class fc1 scheduler s1;
      forwarding-class fc2 scheduler s2;
      forwarding-class fc3 scheduler s3;
      forwarding-class fc4 scheduler s4;
      forwarding-class fc5 scheduler s5;
      forwarding-class fc6 scheduler s6;
```


Copyright © 2019, Juniper Networks, Inc.

```
        forwarding-class fc7 scheduler s7;
    }
  }
}
```

If you are done configuring the device, enter **commit** from configuration mode.

Verification

Confirm that the configuration is working properly.

- Verifying Queue Transmission Rate Oversubscription on page 889

Verifying Queue Transmission Rate Oversubscription

Purpose Verify that the SONET/SDH OC48/STM16 IQE PIC supports 300 percent oversubscription.

Action Configure the queue transmission rates such that they are oversubscribed up to 300 percent.

The sum of transmission rates of all queues can be less than or equal to 300 percent. The sum of transmission rates of all queues in each priority group should be less than or equal to 100 percent.

In this example, Router R0 interfaces have been oversubscribed by 150 percent of the available bandwidth. The sum of transmission rates of all the queues in each of the priority groups are:

- Strict-High—(q0) 25%
- High, Medium-High, Medium-Low—(q1, q2, q3, and q4) 80%
- Low—(q5, q6, and q7) 45%

When the sum of transmission rates of all queues in any of the priority groups exceeds 100 percent, the commit fails.

For example, if the transmission rate of q1 is 30 percent, the sum of the transmission rates of all queues (q1, q2, q3, and q4) in the High-Medium priority group is 110 percent. At the time of commit, the following error is displayed:

```
Total bandwidth allocation for high-med priority queues exceeds 100 percent
for scheduler-map map_ifls
error: configuration check-out failed
```

Meaning When the sum of transmission rates of all queues exceeds 100 percent, a new priority group is created for all Low priority queues. The queue transmission rates in the Low priority group can add up to 100 percent.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Transmission Rate with Intelligent Oversubscription on SONET/SDH OC48/STM16 IQE PICs Overview on page 877

Example: Configuring a CIR and a PIR on SONET/SDH OC48/STM16 IQE Interfaces

On SONET/SDH OC48/STM16 IQE interfaces, you can configure a CIR (guaranteed rate) and a PIR (shaping rate) on a single logical interface. The configured rates are gathered into a traffic control profile. If you configure a traffic control profile with a CIR (guaranteed rate) only, the PIR (shaping rate) is set to the physical interface (port) rate.

NOTE: CIR and PIR are not supported at the queue level.

In the following example, logical unit **0** has a CIR equal to 30 Mbps and a PIR equal to 200 Mbps. Logical unit **1** has a PIR equal to 300 Mbps. Logical unit **2** has a CIR equal to 100 Mbps and a PIR that is unspecified. For logical unit **2**, the software gives the PIR the value of 1 Gbps (equal to the physical interface rate) because the PIR must be equal to or greater than the CIR.

In this example, bandwidth is shared proportionally to the guaranteed rate because at least one logical interface has a guaranteed rate.

```
class-of-service {
  traffic-control-profiles {
    profile1 {
      shaping-rate 200m;
      guaranteed-rate 30m;
      delay-buffer-rate 150m;
      scheduler-map sched-map;
    }
    profile2 {
      shaping-rate 300m;
      delay-buffer-rate 500k;
      scheduler-map sched-map;
    }
    profile3 {
      guaranteed-rate 100m;
      scheduler-map sched-map;
    }
  }
  interfaces {
    se-3/0/0 {
      unit 0 {
        output-traffic-control-profile profile1;
      }
      unit 1 {
        output-traffic-control-profile profile2;
      }
      unit 2 {
        output-traffic-control-profile profile3;
      }
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
  }
}
```

Related Documentation

- Excess Bandwidth Sharing on SONET/SDH OC48/STM16 IQE PICs on page 892

MDRR on SONET/SDH OC48/STM16 IQE PICs

The guaranteed rate (committed information rate) is implemented using modified deficit round-robin (MDRR). MDRR configuration on the SONET/SDH OC48/STM16 IQE PIC is the same as the MDRR configuration on the Enhanced Queuing DPC. For more information about MDRR configuration on the Enhanced Queuing DPC, see "Configuring MDRR on Enhanced Queuing DPCs" on page 913.

Configuring MDRR on SONET/SDH OC48/STM16 IQE PICs

MDRR configuration on the SONET/SDH OC48/STM16 IQE PIC is the same as the MDRR configuration on the Enhanced Queuing DPC. For more information about MDRR configuration on the Enhanced Queuing DPC, see "Configuring MDRR on Enhanced Queuing DPCs" on page 913.

Example: Configuring MDRR on SONET/SDH OC48/STM16 IQE PICs

MDRR configuration on the SONET/SDH OC48/STM16 IQE PIC is same as the MDRR configuration on the Enhanced Queuing DPC. For more information about MDRR configuration on the Enhanced Queuing DPC, see "Configuring MDRR on Enhanced Queuing DPCs" on page 913.

WRED on SONET/SDH OC48/STM16 IQE PICs

Weighted random early detection (WRED) is done at the queue level in the SONET/SDH OC48/STM16 IQE PIC. With WRED, the decision to drop or send the packet is made before the packet is placed in the queue.

WRED configuration on the SONET/SDH OC48/STM16 IQE PIC is the same as the WRED configuration on the Enhanced Queuing DPC. For more information about WRED configuration on the Enhanced Queuing DPC, see "Configuring WRED on Enhanced Queuing DPCs" on page 912.

Related Documentation

- Configuring WRED on SONET/SDH OC48/STM16 IQE PICs on page 891
- Example: Configuring WRED on SONET/SDH OC48/STM16 IQE PICs on page 892

Configuring WRED on SONET/SDH OC48/STM16 IQE PICs

WRED configuration on the SONET/SDH OC48/STM16 IQE PIC is the same as the WRED configuration on the Enhanced Queuing DPC. For more information about WRED

Copyright © 2019, Juniper Networks, Inc.

configuration on the Enhanced Queuing DPC, see "Configuring WRED on Enhanced Queuing DPCs" on page 912.

Example: Configuring WRED on SONET/SDH OC48/STM16 IQE PICs

WRED configuration on the SONET/SDH OC48/STM16 IQE PIC is same as the WRED configuration on the Enhanced Queuing DPC. For more information about WRED configuration on the Enhanced Queuing DPC, see "Configuring WRED on Enhanced Queuing DPCs" on page 912.

Excess Bandwidth Sharing on SONET/SDH OC48/STM16 IQE PICs

Excess bandwidth sharing configuration on SONET/SDH OC48/STM16 IQE PIC is the same as the excess bandwidth sharing on Enhanced Queuing DPC. For more information about excess bandwidth sharing configuration, see "Configuring Excess Bandwidth Sharing" on page 916.

Configuring Excess Bandwidth Sharing on SONET/SDH OC48/STM16 IQE PICs

Excess bandwidth sharing configuration on SONET/SDH OC48/STM16 IQE PIC is the same as the excess bandwidth sharing on Enhanced Queuing DPC. For more information about excess bandwidth sharing configuration, see "Configuring Excess Bandwidth Sharing" on page 916

Egress Rewrite on SONET/SDH OC48/STM16 IQE PICs

The egress rewrite on **inet-precedence**, **dscp**, **dscp-ipv6**, and **exp** is done by the packet forwarding engine (PFE) based on the features supported by the PFE.

Related Documentation

- Applying Rewrite Rules to Output Logical Interfaces on page 401
- Packet Classification on SONET/SDH OC48/STM16 IQE PICs on page 864
- Configuring Translation Tables on SONET/SDH OC48/STM16 IQE PICs on page 866

Configuring Rewrite Rules on SONET/SDH OC48/STM16 IQE PIC

To configure a rewrite rules mapping and associate it with the appropriate forwarding class and code-point alias or bit set, include the **rewrite-rules** statement at the **[edit class-of-service]** hierarchy level:

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


Copyright © 2019, Juniper Networks, Inc.

NOTE: The egress rewrite on the **dscp**, **dscp-ipv6**, **exp**, or **inet-precedence** field is done by the Packet Forwarding Engine based on the features it supports.

Related Documentation

- Mapping CoS Component Inputs to Outputs on page 11
- Egress Rewrite on SONET/SDH OC48/STM16 IQE PICs on page 892

Forwarding Class to Queue Mapping on SONET/SDH OC48/STM16 IQE PICs

Forwarding class to queue mapping is done per data-link connection identifier. For information about configuring forwarding classes and queues, see "Configuring a Custom Forwarding Class for Each Queue" on page 212.

Related Documentation

- Classifying Packets by Egress Interface on page 221

Configuring Forwarding Classes on SONET/SDH OC48/STM16 IQE PIC

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

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

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

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

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring a Custom Forwarding Class for Each Queue on page 212

Configuring Rate Limits on SONET/SDH OC48/STM16 IQE PICs

You can rate-limit all queues on SONET/SDH OC48/STM16 IQE PICs. However, overall you can have only 256 distinct policed rates. Without this limiting, traffic in higher-priority queues can block the transmission of lower-priority packets. If you do not rate-limit queues, higher-priority traffic is always sent before lower-priority traffic, causing the lower-priority queues to "starve," which in turn leads to timeouts and unnecessary resending of packets.

On the SONET/SDH OC48/STM16 IQE PICs, you can rate-limit queues before the packets are queued for output (analogous to policing). All packets exceeding the configured rate limit are dropped, so care is required when establishing this limit. The rate-limit can be configured on the non strict-high queues also.

NOTE: When the queue transmission rates are oversubscribed, the rate-limit option configured on any of the queues uses the configured rate limit values, although the transmission rates are oversubscribed.

To rate-limit queues, include the **transmit-rate** statement with the **rate-limit** option at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level:

```
[edit class-of-service schedulers scheduler-name]
transmit-rate percent percentage rate rate-limit;
Priority priority-level
```

Related Documentation

- Example: Configuring Rate Limits on SONET/SDH OC48/STM16 IQE PICs on page 894

Example: Configuring Rate Limits on SONET/SDH OC48/STM16 IQE PICs

This example limits the transmit rate of a strict-high expedited forwarding queue to 1 megabit per second (Mbps). The scheduler and scheduler map are defined and then applied to the traffic at the **[edit interfaces]** and **[edit class-of-service]** hierarchy levels:

```
[edit class-of-service]
schedulers {
   scheduler-1 {
      transmit-rate 1m rate-limit; # This establishes the limit
      priority strict-high;
   }
}
scheduler-maps {
   scheduler-map-1 {
      forwarding-class expedited-forwarding scheduler scheduler-1;
   }
}
```


Copyright © 2019, Juniper Networks, Inc.

```
[edit interfaces]
s0-2/1/0 {
   per-unit-scheduler;
   encapsulation frame-relay;
   unit 0 {
      dlci 1;
   }
}

[edit class-of-service]
interfaces {
   so-2/1/0 {
      unit 0 {
         scheduler-map scheduler-map-1;
         shaping-rate 2m;
      }
   }
}
```

You can issue the following operational mode commands to verify your configuration (the first shows the rate limit in effect):

- **show class-of-service scheduler-map** *scheduler-map-name*
- **show class-of-service interface** *interface-name*

Related Documentation

- Configuring Rate Limits on SONET/SDH OC48/STM16 IQE PICs on page 894

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 24

Configuring Class of Service on 10-Gigabit Ethernet LAN/WAN PICs with SFP+

- CoS on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview on page 897
- BA and Fixed Classification on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview on page 898
- Example: Configuring IEEE 802.1p BA Classifier on 10-Gigabit Ethernet LAN/WAN PICs on page 899
- DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC with SFP+ on page 900
- Configuring DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC on page 902
- Queuing on 10-Gigabit Ethernet LAN/WAN PICs Properties on page 903
- Mapping Forwarding Classes to CoS Queues on 10-Gigabit Ethernet LAN/WAN PICs on page 904
- Scheduling and Shaping on 10-Gigabit Ethernet LAN/WAN PICs Overview on page 905
- Example: Configuring Shaping Overhead on 10-Gigabit Ethernet LAN/WAN PICs on page 906

CoS on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview

The 10-Gigabit Ethernet LAN/WAN PIC with SFP+ supports intelligent handling of oversubscribed traffic in applications, such as data centers and dense-core uplinks. The 10-Gigabit Ethernet LAN/WAN PIC with SFP+ supports line-rate operation for five 10-Gigabit Ethernet ports from each port group or a total WAN bandwidth of 100 Gbps with Packet Forwarding Engine bandwidth of 50 Gbps.

NOTE: **This PIC has a front panel label with the designation "ETHERNET 10GBASE-SFP+ LAN-WAN" and can also be identified by its model number, PD-5-10XGE-SFPP. It is referred to hereafter as the 10-Gigabit Ethernet LAN/WAN PIC.**

The class-of-service (CoS) configuration for the 10-Gigabit Ethernet LAN/WAN PICs are supported on standalone T640 and T1600 core routers, as well as T640 and T1600 routers in a routing matrix. The 10-Gigabit Ethernet LAN/WAN PICs support behavior aggregate (BA) and fixed classification, weighted round-robin scheduling with two queue

Copyright © 2019, Juniper Networks, Inc.

priorities (low and strict-high), committed and peak information rate shaping on a per-queue basis, and excess information rate configuration for allocation of excess bandwidth.

To configure these features, include the corresponding class-of-service (CoS) statements at the [**edit class-of-service**] hierarchy level. The CoS statements supported on the 10-Gigabit Ethernet LAN/WAN PICs are shown in Table 124 on page 898.

Table 124: CoS Statements Supported on the 10-Gigabit Ethernet LAN/WAN PICs

CoS Statements	Supported
buffer-size	No
drop-profile-map	No
excess-priority	No
excess-rate	Yes
priority	Yes
shaping-rate	Yes
transmit-rate	Yes

Related Documentation

- CoS Features and Limitations on M Series and T Series Routers on page 553
- *Junos OS Network Interfaces Library for Routing Devices*

BA and Fixed Classification on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview

The 10-Gigabit Ethernet LAN/WAN PICs support the following behavior aggregate (BA) classifiers:

- DSCP, DSCP IPv6, or IP precedence—IP packet classification (Layer 3 headers)
- MPLS EXP—MPLS packet classification (Layer 2 headers)
- IEEE 802.1p—Packet classification (Layer 2 headers)
- IEEE 802.1ad—Packet classification for IEEE 802.1ad formats (including DEI bit)

Multiple classifiers can be configured to a single logical interface. However, there are some restrictions on which the classifiers can coexist. For example, the DSCP and IP precedence classifiers cannot be configured on the same logical interface. The DSCP and IP precedence classifiers can coexist with the DSCP IPv6 classifier on the same logical interface. An IEEE 802.1 classifier can coexist with other classifiers and is applicable only if a packet does not match any of the configured classifiers. For information about the supported combinations, see "Applying Behavior Aggregate Classifiers to Logical Interfaces" on page 53.

Copyright © 2019, Juniper Networks, Inc.

If the classifiers are not defined explicitly, then the default classifiers are applied as follows:

- All MPLS packets are classified using the MPLS (EXP) classifier. If there is no explicit MPLS (EXP) classifier, then the default MPLS (EXP) classifier is applied.
- All IPv4 packets are classified using the IP precedence and DSCP classifiers. If there is no explicit IP precedence or DSCP classifier, then the default IP precedence classifier is applied.
- All IPv6 packets are classified using a DSCP IPv6 classifier. If there is no explicit DSCP IPv6 classifier, then the default DSCP IPv6 classifier is applied.
- If the IEEE 802.1p classifier is configured and a packet does not match any explicitly configured classifier, then the IEEE 802.1p classifier is applied.

The fixed classification matches the traffic on a logical interface level. The following example classifies all traffic on logical unit zero to the queue corresponding to assured forwarding.

```
[edit class-of-service interfaces xe-0/1/2 unit 0]
forwarding-class fc-af11;
```

NOTE: The 10-Gigabit Ethernet LAN/WAN PICs do not support multifield classification. However, the multifield classification can be done at the Packet Forwarding Engine using the firewall filters, which overrides the classification done at the PIC level. The multifield classification at the Packet Forwarding Engine occurs after the PIC handles the oversubscribed traffic.

Example: Configuring IEEE 802.1p BA Classifier on 10-Gigabit Ethernet LAN/WAN PICs

To configure an IEEE 802.1p behavior aggregate (BA) classifier on the 10-Gigabit Ethernet LAN/WAN PICs, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service classifiers]
ieee-802.1 classifier-name {
   forwarding-class fc-nc2 {
      loss-priority low code-points [111];
   }
   forwarding-class fc-nc1 {
      loss-priority low code-points [110];
   }
   forwarding-class fc-af12 {
      loss-priority low code-points [101];
   }
   forwarding-class fc-af11 {
      loss-priority low code-points [100];
   }
   forwarding-class fc-ef1 {
```

Copyright © 2019, Juniper Networks, Inc.

```
    loss-priority low code-points [011];
  }
  forwarding-class fc-ef {
    loss-priority low code-points [010];
  }
  forwarding-class fc-be1 {
    loss-priority low code-points [001];
  }
  forwarding-class fc-be {
    loss-priority low code-points [000];
  }
}
[edit class-of-service interfaces xe-0/1/2 unit 0]
classifiers {
  ieee-802.1 classifier-name;
}
```

NOTE: The 10-Gigabit Ethernet LAN/WAN PICs do not support queuing at the logical interface level. However, classifiers can be configured on individual logical interfaces. The same classifier can be configured on multiple logical interfaces.

Related Documentation

- BA and Fixed Classification on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview on page 898

DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC with SFP+

The 10-Gigabit Ethernet LAN/WAN PIC with SFP+ (Model Number: PD-5-10XGE-SFPP) in T640 and T1600 standalone routers and TX Matrix and TX Matrix Plus routing matrices supports 6-bit DSCP rewrite (IPv4 and IPv6) functionality. The following DSCP rewrite features are supported:

- Full 6-bit DSCP rewrite
- Independent rewrite for DSCPv4 and DSCPv6 simultaneously on the same logical interface
- Four tables per PIC for DSCPv4 and DSCPv6, respectively
- Rewrite based on queue number rather than forwarding class. Queues are mapped to a forwarding class by using the global **forwarding-class** configuration on the router.
- Ability to bind multiple (maximum of all) logical interfaces on the PIC to the same rewrite table.
- Ability of DSCP rewrite on the PIC to configure, by default, code-point 000000 if you do not specify a classifier in the **rewrite-rules** statement.

Copyright © 2019, Juniper Networks, Inc.

NOTE:

The 10-Gigabit Ethernet LAN/WAN PIC with SFP+ (P/N: PD-5-10XGE-SFPP), when used in T640 and T1600 standalone routers, and T640 and T1600 routers in TX Matrix and TX Matrix Plus routing matrices, has the following known limitations:

- DSCP rewrite on the PIC does not support distinct DSCP code-point rewrites if multiple forwarding classes (FC) are configured to map to the same queue in the "forwarding-class" configuration.
- The PIC can perform DSCP rewrite based on three PLP values, unlike four PLP values by the Packet Forwarding Engine.
- The protocol option is not supported in the following DSCP rewrite rule configuration:

  ```
  [edit class-of-service interfaces interface-name unit logical-unit-number]
  rewrite-rules {
    dscp (rewrite-name | <default>) protocol <protocol-types>;
  }
  ```

- The PIC has the ability to parse a packet with up to two VLAN tags. However, the following conditions apply when DSCP rewrite is enabled:
  - The PIC supports DSCP rewrite only for untagged and single VLAN tagged packets.
  - For DSCP rewrite in conjunction with VLAN rewrite push operations, the PIC can push only one tag if the packet is untagged.
  - If the packet has more than one VLAN tag (either because it was double tagged or because additional tags were pushed as part of a VLAN rewrite), then DSCP rewrite is not executed.
- Configuration of DSCP rewrite rules on the PIC overwrites the DSCP value coming from the Routing Engine for host-generated traffic. The behavior is as follows:
  - If the packet's forwarding class and packet loss priority (PLP) match the DSCP rewrite rule on the PIC, then the DSCP **code-point** rewritten by the **host-outbound-traffic** statement is overwritten by the PIC's DSCP rewrite with the corresponding DSCP **code-point** configured in the rewrite rule.
  - If the packet's forwarding class and PLP do not match any DSCP rewrite rule on the PIC, then the DSCP **code-point** rewritten by the **host-outbound-traffic** statement is overwritten by the PIC's DSCP rewrite as 6b'000000.

  This behavior is different from DSCP rewrites done in the Packet Forwarding Engine for other PICs. In those cases, the Packet Forwarding Engine processing is bypassed for host-generated packets and hence the DSCP

Copyright © 2019, Juniper Networks, Inc.

set in the Routing Engine for host-generated packets is not overwritten in the Packet Forwarding Engine or PIC.

- If multiple forwarding classes map to the same queue, then the last forwarding class that maps to the same queue is picked and its **code-point** is used for DSCP rewrite.
- If both **medium-high** and **medium-low** PLP values are configured in the rewrite rule and if their rewrite **code-points** are different, then the **code-point** associated with **medium-high** is used for rewrite for both **medium-high** and **medium-low** packets on that logical interface. If only one of the PLP values (either **medium-high** or **medium-low**) is configured, then its corresponding **code-point** is used for rewrite for both **medium-high** and **medium-low** packets on that logical interface.

NOTE: A system error message can result if a configuration that conflicts with these limitations is committed or used .

Related Documentation

- Configuring DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC on page 902
- **dscp on page 1093**
- **dscp-ipv6 on page 1096**
- **forwarding-class on page 1134**
- **rewrite-rules on page 1249**
- Understanding DSCP Classification for VPLS on page 75
- Default DSCP and DSCP IPv6 Classifiers on page 41

Configuring DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC

To configure DSCP rewrite, use the **rewrite-rules** statement at the **class-of-service interfaces** ***interface-name*** **unit** ***logical-unit-number*** hierarchy level, as shown in the following configuration example:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
rewrite-rules {
   dscp (rewrite-name | <default>);
   dscp-ipv6 (rewrite-name | <default>);
   exp (rewrite-name | <default>) protocol <protocol-types>;
   exp-push-push-push <default>;
   exp-swap-push-push <default>;
   ieee-802.1 (rewrite-name | <default>) vlan-tag (outer | outer-and-inner);
   inet-precedence (rewrite-name | <default>) protocol <protocol-types>;
}
```


Copyright © 2019, Juniper Networks, Inc.

To configure DSCP rewrite rules, use the **rewrite-rules** statement's (<dscp> | <dscp-ipv6>) option's subordinate rewrite rules statements at the **edit class-of-service** hierarchy level, as shown in the following configuration example:

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

Related Documentation

- DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC with SFP+ on page 900
- dscp on page 1093
- dscp-ipv6 on page 1096
- forwarding-class on page 1134
- rewrite-rules on page 1249
- Understanding DSCP Classification for VPLS on page 75
- Default DSCP and DSCP IPv6 Classifiers on page 41

Queuing on 10-Gigabit Ethernet LAN/WAN PICs Properties

The 10-Gigabit Ethernet LAN/WAN PICs have the following features to support queuing:

- Committed and peak information rate shaping on a per-queue basis
- Excess information rate configuration for allocation of excess bandwidth
- Ingress queuing based on behavior aggregate (BA) classification
- Egress queuing at the Packet Forwarding Engine and at the PIC level

  The Packet Forwarding Engine egress queues are shared by two physical interfaces in a port group.
- Weighted round-robin (WRR) scheduling with two queue priorities (low and strict-high)
- Two special queues available in ingress, one per physical interface, called *control queues*

  Layer 2 and Layer 3 control protocol packets (OSPF, OSPF3, VRRP, IGMP, RSVP, PIM, BGP, BFD, LDP, ISIS, RIP, RIPV6, LACP, ARP, IPv6 NDP, CFM, and LFM) are mapped to the control queue. In the control queue, these packets are not dropped even if there is oversubscription or congestion on a port group.

NOTE: The control queue is rate-limited to 2 Mbps per physical interface. The packets in excess of 2 Mbps are dropped and accounted for.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation
- Mapping Forwarding Classes to CoS Queues on 10-Gigabit Ethernet LAN/WAN PICs on page 904

Mapping Forwarding Classes to CoS Queues on 10-Gigabit Ethernet LAN/WAN PICs

The 10-Gigabit Ethernet LAN/WAN PICs support eight CoS queues per port in the egress direction. To map forwarding classes to the eight CoS queues in egress, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service forwarding-classes] {
   class fc-be queue-num 0;
   class fc-be1 queue-num 1;
   class fc-ef queue-num 2;
   class fc-ef1 queue-num 3;
   class fc-af11 queue-num 4;
   class fc-af12 queue-num 5;
   class fc-nc1 queue-num 6;
   class fc-nc2 queue-num 7;
}
```

CAUTION: 10-Gigabit Ethernet LAN/WAN PICs do not support more than eight forwarding classes. If you define more than eight forwarding classes, excess forwarding classes can get mapped to queues with undefined schedulers.

The 10-Gigabit Ethernet LAN/WAN PICs support four ingress queues per physical interface. The PICs use restricted-queues configuration to map multiple forwarding classes to the four queues. There are no queues at the logical interface level. In the following example, two forwarding classes are mapped to one queue.

```
[edit class-of-service restricted-queues] {
   forwarding-class fc-be queue-num 0;
   forwarding-class fc-be1 queue-num 0;
   forwarding-class fc-ef queue-num 1;
   forwarding-class fc-ef1 queue-num 1;
   forwarding-class fc-af11 queue-num 2;
   forwarding-class fc-af12 queue-num 2;
   forwarding-class fc-nc1 queue-num 3;
   forwarding-class fc-nc2 queue-num 3;
}
```

Related Documentation
- Queuing on 10-Gigabit Ethernet LAN/WAN PICs Properties on page 903
- Understanding How Forwarding Classes Assign Classes to Output Queues on page 205
- Configuring a Custom Forwarding Class for Each Queue on page 212
- **forwarding-classes on page 1139**

Copyright © 2019, Juniper Networks, Inc.

Scheduling and Shaping on 10-Gigabit Ethernet LAN/WAN PICs Overview

The 10-Gigabit Ethernet LAN/WAN PIC has ten 10-Gigabit Ethernet ports providing 100 Gbps of WAN bandwidth and 50 Gbps of Packet Forwarding Engine bandwidth. On the ingress side of the 10-Gigabit Ethernet LAN/WAN PIC, two consecutive physical interfaces on the PICs are grouped together into a port group and are serviced by a single scheduler. The port groups are as shown in Table 125 on page 905:

Table 125: Port Groups on 10-Gigabit Ethernet LAN/WAN PICs

Port Group	Mapped Ports
Group 1	xe-*fpc*/*pic*/0 xe-*fpc*/*pic*/1
Group 2	xe-*fpc*/*pic*/2 xe-*fpc*/*pic*/3
Group 3	xe-*fpc*/*pic*/4 xe-*fpc*/*pic*/5
Group 4	xe-*fpc*/*pic*/6 xe-*fpc*/*pic*/7
Group 5	xe-*fpc*/*pic*/8 xe-*fpc*/*pic*/9

The two physical interfaces in a port group share 10 Gbps bandwidth towards the Packet Forwarding Engine. A scheduler has eight class-of-service (CoS) queues and two control queues. On the ingress side of the 10-Gigabit Ethernet LAN/WAN PIC, the eight CoS queues are split four plus four for the two physical interfaces. Thus, the 10-Gigabit Ethernet LAN/WAN PIC supports four ingress queues and eight egress queues per physical interface.

At the ingress side of the 10-Gigabit Ethernet LAN/WAN PIC, multiple forwarding classes can be mapped to one queue using the restricted-queue configuration. When creating a scheduler-map for the ingress queues, only one forwarding class should be chosen from the multiple forwarding classes that map to the same queue. Then, the scheduler-map can be specified using the **set class-of-service scheduler-maps** ***map-name*** **forwarding-class** ***class-name*** **scheduler** ***scheduler*** command.

The 10-Gigabit Ethernet LAN/WAN PICs manage packet buffering internally and no configuration is required.

NOTE: **The delay-bandwidth buffering configuration is not supported on the 10-Gigabit Ethernet LAN/WAN PICs.**

Copyright © 2019, Juniper Networks, Inc.

Example: Configuring Shaping Overhead on 10-Gigabit Ethernet LAN/WAN PICs

By default, the 10-Gigabit Ethernet LAN/WAN PIC uses 20 bytes as the shaping overhead. This includes 8 bytes preamble and 12 bytes interpacket gap (IPG) in shaper operations. To exclude this overhead, it should be configured as –20 bytes. The shaping overhead value can be set between 0 and 31 bytes, as shown in the following example. This range translates to a CLI range of –20 to 11 bytes for the shaping overhead configuration.

```
show chassis
   fpc 6 {
   pic 0 {
      traffic-manager {
      ingress-shaping-overhead –20;
      egress-shaping-overhead –20;
      }
   }
}
```

NOTE: When the configuration for the overhead bytes on a PIC are changed, the PIC is taken offline and then brought back online. In addition, the configuration in the CLI is on a per-PIC basis, and thus, applies to all the ports on the PIC.

Related Documentation

- Scheduling and Shaping on 10-Gigabit Ethernet LAN/WAN PICs Overview on page 905

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 25

Configuring Class of Service on Enhanced Queuing DPCs

- Enhanced Queuing DPC CoS Properties on page 908
- Configuring Rate Limits on Enhanced Queuing DPCs on page 910
- Configuring WRED on Enhanced Queuing DPCs on page 912
- Configuring MDRR on Enhanced Queuing DPCs on page 913
- Configuring Excess Bandwidth Sharing on page 916
- Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs on page 920
- Configuring Customer VLAN (Level 3) Shaping on Enhanced Queuing DPCs on page 922
- Simple Filters Overview on page 924
- Configuring Simple Filters on Enhanced Queuing DPCs on page 925
- Configuring a Simple Filter on page 926

Copyright © 2019, Juniper Networks, Inc.

Enhanced Queuing DPC CoS Properties

On a Juniper Networks MX Series 5G Universal Routing Platform with Enhanced Queuing Dense Port Concentrators (DPCs), you can configure schedulers and queues. You can configure 15 VLAN sets per Gigabit Ethernet (1G) port and 255 VLAN sets per 10-Gigabit Ethernet (10G) port. The Enhanced Queuing DPC performs priority propagation from one hierarchy level to another and drop statistics are available on the Enhanced Queuing DPC per color per queue instead or just per queue.

NOTE: The Enhanced Queuing DPC (EQ DPC) does not support BA classification for packets received from a Layer 3 routing interface or a virtual routing and forwarding (VRF) interface and routed to an integrated routing and bridging interface (IRB) to reach the remote end of a pseudowire connection. The EQ DPC also does not support BA classification for Layer 2 frames received from a Virtual Private LAN Service (VPLS) pseudowire connection from a remote site and routed to a Layer 3 routing interface through an IRB interface.

Juniper Networks MX Series 5G Universal Routing Platforms with Enhanced Queuing DPCs have Packet Forwarding Engines that can support up to 515 MB of frame memory, and packets are stored in 512-byte frames. Table 126 on page 908 compares the major properties of the Intelligent Queuing 2 (IQ2) PIC and the Packet Forwarding Engine within the Enhanced Queuing DPC.

Table 126: IQ2 PIC and Enhanced Queuing DPC Compared

Feature	IQ2 PIC	Packet Forwarding Engine Within Enhanced Queuing DPC
Number of usable queues	8,000	16,000
Number of shaped logical interfaces	1,000 with 8 queues each.	2,000 with 8 queues each, or 4,000 with 4 queues each.
Number of hardware priorities	2	4
Priority propagation	No	Yes
Dynamic mapping	No: schedulers/port are fixed.	Yes: schedulers/port are not fixed.
Drop statistics	Per queues	Per queue per color (PLP high, low)

In addition, the Enhanced Queuing DPC features support for hierarchical weighted random early detection (WRED) and enhanced queuing on aggregated Ethernet interfaces with link protection as well.

Copyright © 2019, Juniper Networks, Inc.

The Enhanced Queuing DPC supports the following hierarchical scheduler characteristics:

- Shaping at the physical interface level
- Shaping and scheduling at the service VLAN interface set level
- Shaping and scheduling at the customer VLAN logical interface level
- Scheduling at the queue level

VLAN (Level 3) shaping on a 10-Gigabit Ethernet MX Series Enhanced Queuing DPC differs from the VLAN (Level3) shaping on a 1-Gigabit Ethernet Enhanced Queuing DPC. To use the VLAN (Level 3) shaping on a 10-Gigabit Ethernet MX Series Enhanced Queuing DPC, configure an interface set at the **[edit interfaces interface-set]** hierarchy level. The interface set configuration is not required for configuring a 1-Gigabit Ethernet VLANs on the same Enhanced Queuing DPC.

The Enhanced Queuing DPC supports the following features for scalability:

- 16,000 queues per Packet Forwarding Engine
- 4 Packet Forwarding Engines per DPC
  - 4000 schedulers at logical interface level (Level 3) with 4 queues each
  - 2000 schedulers at logical interface level (Level 3) with 8 queues each
- 255 schedulers at the interface set level (Level 2) per 1-port Packet Forwarding Engine on a 10-Gigabit Ethernet DPC
- 15 schedulers at the interface set level (Level 2) per 10-port Packet Forwarding Engine on a 1-Gigabit Ethernet DPC
- About 400 milliseconds of buffer delay (this varies by packet size and if large buffers are enabled)
- 4 levels of priority (strict-high, high, medium, and low)

NOTE: Including the **transmit-rate** *rate* **exact** statement at the **[edit class-of-service schedulers** *scheduler-name***]** hierarchy level is not supported on Enhanced Queuing DPCs on MX Series routers.

The way that the Enhanced Queuing DPC maps a queue to a scheduler depends on whether 8 queues or 4 queues are configured. By default, a scheduler at level 3 has 4 queues. Level 3 scheduler X controls queue X*4 to X*4+3, so that scheduler 100 (for example) controls queues 400 to 403. However, when 8 queues per scheduler are enabled, the odd numbered schedulers are disabled, allowing twice the number of queues per subscriber as before. With 8 queues, level 3 scheduler X controls queue X*4 to X*4+7, so that scheduler 100 (for example) now controls queues 400 to 407.

You configure the **max-queues-per-interface** statement to set the number of queues at **4** or **8** at the FPC level of the hierarchy. Changing this statement results in a restart of the DPC. For more information about the **max-queues-per-interface** statement, see the *Junos OS Network Interfaces Library for Routing Devices*.

Copyright © 2019, Juniper Networks, Inc.

The Enhanced Queuing DPC maps level 3 (customer VLAN) schedulers in groups to level 2 (service VLAN) schedulers. Sixteen contiguous level 3 schedulers are mapped to level 2 when 4 queues are enabled, and 8 contiguous level 3 schedulers are mapped to level 2 when 8 queues are enabled. All of the schedulers in the group should use the same queue priority mapping. For example, if the queue priorities of one scheduler are high, medium, low, and low, then all members of this group should have the same queue priority.

Mapping of a group at level 3 to level 2 can be done at any time. However, a group at level 3 can only be unmapped from a level 2 scheduler only if all the schedulers in the group are free. Once unmapped, a level 3 group can be remapped to any level 2 scheduler. There is no restriction on the number of level 3 groups that can be mapped to a particular level 2 scheduler. There can be 256 level 3 groups, but fragmentation of the scheduler space can reduce the number of schedulers available. In other words, there are scheduler allocation patterns that might fail even though there are free schedulers.

In contrast to level-3-to-level-2 mapping, the Enhanced Queuing DPC maps level 2 (service VLAN) schedulers in a fixed mode to level 1 (physical interface) schedulers. On 40-port Gigabit Ethernet DPCs, there are 16 level 1 schedulers, and 10 of these are used for the physical interfaces. There are 256 level 2 schedulers, or 16 per level 1 scheduler. A level 1 scheduler uses level schedulers X*16 through X*16+15. So level 1 scheduler 0 uses level 2 schedulers 0 through 15, level 1 scheduler 1 uses level 2 schedulers 16 through 31, and so on. On 4-port 10-Gigabit Ethernet PICs, there is one level 1 scheduler for the physical interface, and 256 level 2 schedulers are mapped to the single level 1 scheduler.

The maximum number of level 3 (customer VLAN) schedulers that can be used is 4076 (4 queues) or 2028 (8 queues) for the 10-port Gigabit Ethernet Packet Forwarding Engine and 4094 (4 queues) or 2046 (8 queues) for the 10-Gigabit Ethernet Packet Forwarding Engine.

Enhanced Queuing is supported on aggregated Ethernet (AE) interfaces with two links in link protection mode. However, only one link in the AE bundle can be active at a time. Traffic is shaped independently on the two links, but the member's links do not need to reside in the same Packet Forwarding Engine or the same DPC. Finally, shared schedulers are not supported on the Enhanced Queuing DPC (use hierarchical schedulers to group logical interfaces).

Related Documentation

- Configuring Customer VLAN (Level 3) Shaping on Enhanced Queuing DPCs on page 922

Configuring Rate Limits on Enhanced Queuing DPCs

You can rate-limit the strict-high and high queues on the Enhanced Queuing DPC. Without rate limits, traffic in higher-priority queues can block the transmission of lower-priority packets. Unless limited, higher-priority traffic is always sent before lower-priority traffic, causing the lower-priority queues to "starve" and cause timeouts and unnecessarily resent packets.

On the Enhanced Queuing DPC, you can rate-limit queues before the packets are queued for output. All packets exceeding the configured rate limit are dropped, so care is required

Copyright © 2019, Juniper Networks, Inc.

when establishing this limit. This model is also supported on IQ2 PICs. For more information about configuring CoS on IQ2 PICs, see "CoS on Enhanced IQ2 PICs Overview" on page 794.

NOTE: Rate limiting is implemented differently on Enhanced Queuing DPCs and non-queuing Packet Forwarding Engines. On Enhanced Queuing DPCs, rate limiting is implemented using a single-rate two-color policer. On nonqueuing Packet Forwarding Engines, rate limiting is achieved by shaping the queue to the transmit rate and keeping the queue delay buffers small to prevent too many packets from being queued after the shaping rate is reached.

To rate-limit queues, include the **transmit-rate** statement with the **rate-limit** option at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level:

```
[edit class-of-service schedulers scheduler-name]
transmit-rate rate rate-limit;
```

The following example limits the transmit rate of a strict-high expedited-forwarding queue to 1 Mbps. The scheduler and scheduler map are defined, and then applied to the traffic at the **[edit interfaces]** and **[edit class-of-service]** hierarchy levels:

```
[edit class-of-service]
schedulers {
   scheduler-1 {
      transmit-rate 1m rate-limit; # This establishes the limit
      priority strict-high;
   }
}
scheduler-maps {
   scheduler-map-1 {
      forwarding-class expedited-forwarding scheduler scheduler-1;
   }
}

[edit interfaces]
s0-2/2/0 {
   per-unit-scheduler;
   encapsulation frame-relay;
   unit 0 {
      dlci 1;
   }
}

[edit class-of-service]
interfaces {
   so-2/2/0 {
      unit 0 {
         scheduler-map scheduler-map-1;
         shaping-rate 2m;
      }
   }
```

Copyright © 2019, Juniper Networks, Inc.

```
}
```

You can issue the following operational mode commands to verify your configuration (the first shows the rate limit in effect):

- **show class-of-service scheduler-map** *scheduler-map-name*
- **show class-of-service interface** *interface-name*

You can issue the **show interfaces queue** *interface-name* command to view the number of packets dropped at an interface. The output of the **show interfaces queue** *interface-name* command always displays the rate-limit counter fields whether or not rate limiting is configured on the queue. Rate-limit counters are displayed in two columns. The first column is the consolidated count of the packets dropped and the second column is the real-time count of the packets dropped.

Rate-limit packet drop counters display the value 0 when rate limiting is not configured on the queue or when the queue does not have rate-limit packet drops even with rate limiting configured.

Rate-limit packet drop counters display meaningful values in both columns when the queue has rate-limit packet drops. However, when rate limiting is not happening in real time but has occurred earlier, the first column displays the consolidated count and the second column displays the value 0.

You can clear the packet drop statistics by using the **clear interface statistics** *interface-name* command.

Configuring WRED on Enhanced Queuing DPCs

Shaping to drop out-of-profile traffic is done on the Enhanced Queuing DPC at all levels but the queue level. However, weighed random early detection (WRED) is done at the queue level with much the same result. With WRED, the decision to drop or send the packet is made before the packet is placed in the queue.

WRED shaping on the Enhanced Queuing DPC is similar to the IQ2 PIC, but involves only two levels, not 64. The probabilistic drop region establishes a minimum and a maximum queue depth. Below the minimum queue depth, the drop probability is 0 (send). Above the maximum level, the drop probability is 100 (certainty).

There are four drop profiles associated with each queue. These correspond to each of four loss priorities (low, medium-low, medium-high, and high). Sixty-four sets of four drop profiles are available (32 for ingress and 32 for egress). In addition, there are eight WRED scaling profiles in each direction.

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

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


Copyright © 2019, Juniper Networks, Inc.

```
    }
}
```

The following example is an Enhanced Queuing DPC drop profile for expedited forwarding traffic:

```
[edit class-of-service drop-profiles]
drop-ef {
    fill-level 20 drop-probability 0; # Minimum Q depth
    fill-level 100 drop-probability 100; # Maximum Q depth
}
```

Note that only two fill levels can be specified for the Enhanced Queuing DPC. You can configure the **interpolate** statement, but only two fill levels are used. The **delay-buffer-rate** statement in the traffic control profile determines the maximum queue size. This delay buffer rate is converted to a packet delay buffers, where one buffer is equal to 512 bytes. For example, at 10 Mbps, the Enhanced Queuing DPC allocates 610 delay buffers when the delay-buffer rate is set to 250 milliseconds. The WRED threshold values are specified in terms of absolute buffer values.

The WRED scaling factor multiples all WRED thresholds (both minimum and maximum) by the value specified. There are eight values in all: 1, 2, 4, 8, 16, 32, 64, and 128. The WRED scaling factor is chosen to best match the user-configured drop profiles. This is done because the hardware supports only certain values of thresholds (all values must be a multiple of 16). So if the configured value of a threshold is 500 (for example), the multiple of 16 is 256 and the scaling factor applied is 2, making the value 512, which allows the value of 500 to be used. If the configured value of a threshold is 1500, the multiple of 16 is 752 and the scaling factor applied is 2, making the value 1504, which allows the value of 1500 to be used

Hierarchical RED is used to support the oversubscription of the delay buffers (WRED is only configured at the queue, physical interface, and PIC level). Hierarchical RED works with WRED as follows:

- If any level accepts the packet (the queue depth is less than the minimum buffer level), then this level accepts the packet.
- If any level probabilistically drops the packet, then this level drops the packet.

However, these rules might lead to the accepting of packets under loaded conditions which might otherwise have been dropped. In other words, the logical interface accepts packets if the physical interface is not congested.

Related Documentation

- Shaping Granularity Values for Enhanced Queuing Hardware on page 798
- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351.

Configuring MDRR on Enhanced Queuing DPCs

The guaranteed rate (CIR) at the interface set level is implemented using modified deficit round-robin (MDRR). The Enhanced Queuing DPC hardware provides four levels of strict

Copyright © 2019, Juniper Networks, Inc.

priority. There is no restriction on the number of queues for each priority. MDRR is used among queues of the same priority. Each queue has one priority when it is under the guaranteed rate and another priority when it is over the guaranteed rate but under the shaping rate (PIR). The Enhanced Queuing DPC hardware implements the priorities with 256 service profiles. Each service profile assigns eight priorities for eight queues. One set is for logical interfaces under the guaranteed rate and another set is for logical interfaces over the guaranteed rate but under the shaping rate. Each service profile is associated with a group of 16 level 3 schedulers, so there is a unique service profile available for all 256 groups at level 3, giving 4096 logical interfaces.

The Junos OS provides three priorities for traffic under the guaranteed rate and one reserved priority for traffic over the guaranteed rate that is not configurable. The Junos OS provides three priorities when there is no guaranteed rate configured on any logical interface.

The relationship between Junos OS priorities and the Enhanced Queuing DPC hardware priorities below and above the guaranteed rate (CIR) is shown in Table 127 on page 914.

Table 127: Junos OS Priorities Mapped to Enhanced Queuing DPC Hardware Priorities

Junos OS Priority	Enhanced Queuing DPC Hardware Priority Below Guaranteed Rate	Enhanced Queuing DPC Hardware Priority Above Guaranteed Rate
Strict-high	High	High
High	High	Low
Medium-high	Medium-high	Low
Medium-low	Medium-high	Low
Low	Medium-low	Low

To configure MDRR, configure a scheduler at the **[edit class-of-service schedulers]** hierarchy level:

```
[edit class-of-service schedulers]
scheduler-name {
   buffer-size (seconds | percent percentage | remainder | temporal microseconds);
   priority priority-level;
   transmit-rate (percent percentage | rate | remainder) <exact | rate-limit>;
}
```

The following example creates two schedulers for MDRR:

```
[edit class-of-service schedulers]
best-effort-scheduler {
   transmit-rate percent 30; # if no shaping rate
   buffer-size percent 30;
   priority high;
}
expedited-forwarding-scheduler {
```


Copyright © 2019, Juniper Networks, Inc.

```
    transmit-rate percent 40; # if no shaping rate
    buffer-size percent 40;
    priority strict-high;
}
```

NOTE: The use of both shaping rate and a guaranteed rate at the interface set level (level 2) is not supported.

MDRR is provided at three levels of the scheduler hierarchy of the Enhanced Queuing DPC with a granularity of 1 through 255. There are 64 MDRR profiles at the queue level, 16 at the interface set level, and 32 at the physical interface level.

Queue transmit rates are used for queue level MDRR profile weight calculation. The queue MDRR weight is calculated differently based on the mode set for sharing excess bandwidth. If you configure the **equal** option for excess bandwidth, then the queue MDRR weight is calculated as:

Queue weight = (255 * Transmit-rate-percentage) / 100

If you configure the **proportional** option for excess bandwidth, which is the default, then the queue MDRR weight is calculated as:

Queue weight = Queue-transmit-rate / Queue-base-rate, where

Queue-transmit-rate = (Logical-interface-rate * Transmit-rate-percentage) / 100, and

Queue-base-rate = Excess-bandwidth-proportional-rate / 255

To configure the way that the Enhanced Queuing DPC should handle excess bandwidth, configure the **excess-bandwidth-share** statement at the **[edit interface-set *interface-set-name*]** hierarchy level. By default, the excess bandwidth is set to **proportional** with a default value of 32.64 Mbps. In this mode, the excess bandwidth is shared in the ratio of the logical interface shaping rates. If set to **equal**, the excess bandwidth is shared equally among the logical interfaces.

This example sets the excess bandwidth sharing to proportional at a rate of 100 Mbps with a shaping rate of 80 Mbps.

```
[edit interface-set example-interface-set]
excess-bandwidth-share proportional 100m;
output-traffic-control-profile PIR-80Mbps;
```

Shaping rates established at the logical interface level are used to calculate the MDRR weights used at the interface set level. The 16 MDRR profiles are set to initial values, and the closest profile with rounded values is chosen. By default, the physical port MDRR weights are preset to the full bandwidth on the interface.

Copyright © 2019, Juniper Networks, Inc.

Configuring Excess Bandwidth Sharing

When using the Enhanced Queuing DPC on an MX Series router, there are circumstances when you should configure excess bandwidth sharing and minimum logical interface shaping. This section details some of the guidelines for configuring excess bandwidth sharing.

- Excess Bandwidth Sharing and Minimum Logical Interface Shaping on page 916
- Selecting Excess Bandwidth Sharing Proportional Rates on page 916
- Mapping Calculated Weights to Hardware Weights on page 917
- Allocating Weight with Only Shaping Rates or Unshaped Logical Interfaces on page 918
- Sharing Bandwidth Among Logical Interfaces on page 919

Excess Bandwidth Sharing and Minimum Logical Interface Shaping

The default excess bandwidth sharing proportional rate is 32.65 Mbps (128 Kbps x 255). In order to have better weighed fair queuing (WFQ) accuracy among queues, the shaping rate configured should be larger than the excess bandwidth sharing proportional rate. Some examples are shown in Table 128 on page 916.

Table 128: Shaping Rates and WFQ Weights

Shaping Rate	Configured Queue Transmit Rate	WFQ Weight	Total Weights
10 Mbps	(30, 40, 25, 5)	(22, 30, 20, 4)	76
33 Mbps	(30, 40, 25, 5)	(76, 104, 64, 13)	257
40 Mbps	(30, 40, 25, 5)	(76, 104.64, 13)	257

With a 10-Mbps shaping rate, the total weights are 76. This is divided among the four queues according to the configured transmit rate. Note that when the shaping rate is larger than the excess bandwidth sharing proportional rate of 32.65 Mbps, the total weights on the logical interface are 257 and the WFQ accuracy is the same.

Selecting Excess Bandwidth Sharing Proportional Rates

A good excess bandwidth sharing proportional rate to configure is to choose the largest CIR (guaranteed rate) among all the logical interfaces (units). If the logical units have PIRs (shaping rates) only, then choose the largest PIR rate. However, this is not ideal if a single logical interface has a large weighed round-robin (WRR) rate. This can skew the distribution of traffic across the queues of the other logical interfaces. To avoid this issue, set the excess bandwidth sharing proportional rate to a lower value on the logical interfaces where the WRR rates are concentrated. This improves the bandwidth sharing accuracy among the queues on the same logical interface. However, the excess bandwidth sharing for the logical interface with the larger WRR rate is no longer proportional.

As an example, consider five logical interfaces on the same physical port, each with four queues, all with only PIRs configured and no CIRs. The WRR rate is the same as the PIR

Copyright © 2019, Juniper Networks, Inc.

for the logical interface. The excess bandwidth is shared proportionally with a rate of 40 Mbps. The traffic control profiles for the logical interfaces are shown in Table 129 on page 917.

Table 129: Sample Shaping Rates and WFQ Weights

Shaping Rate	Configured Queue Transmit Rate	WFQ Weight	Total Weights
(Unit 0) 10 Mbps	(95, 0, 0, 5)	(60, 0, 0, 3)	63
(Unit 1) 20 Mbps	(25, 25, 25, 25)	(32, 32, 32, 32)	128
(Unit 2) 40 Mbps	(40, 30, 20, 10)	(102, 77, 51, 26)	255
(Unit 3) 200 Mbps	(70, 10, 10, 10)	(179, 26, 26, 26)	255
(Unit 4) 2 Mbps	(25, 25, 25, 25)	(5, 5, 5, 5)	20

Even though the maximum transmit rate for the queue on logical interface unit 3 is 200 Mbps, the excess bandwidth sharing proportional rate is kept at a much lower value. Within a logical interface, this method provides a more accurate distribution of weights across queues. However, the excess bandwidth is now shared equally between unit 2 and unit 3 (total weight of each = 255).

Mapping Calculated Weights to Hardware Weights

The calculated weight in a traffic control profile is mapped to hardware weight, but the hardware only supports a limited WFQ profile. The weights are rounded to the nearest hardware weight according to the values in Table 130 on page 917.

Table 130: Rounding Configured Weights to Hardware Weights

Traffic Control Profile Number	Number of Traffic Control Profiles	Weights	Maximum Error
1–16	16	1–16 (interval of 1)	50.00%
17–29	13	18–42 (interval of 2)	6.25%
30–35	6	45–60 (interval of 3)	1.35%
36–43	8	64–92 (interval of 4)	2.25%
44–49	6	98–128 (interval of 6)	3.06%
50–56	7	136–184 (interval of 8)	3.13%
57–62	6	194–244 (interval of 10)	2.71%
63–63	1	255–255 (interval of 11)	2.05%

Copyright © 2019, Juniper Networks, Inc.

From the table, as an example, the calculated weight of 18.9 is mapped to a hardware weight of 18, because 18 is closer to 18.9 than 20 (an interval of 2 applies in the range 18–42).

Allocating Weight with Only Shaping Rates or Unshaped Logical Interfaces

Logical interfaces with only shaping rates (PIRs) or unshaped logical interfaces (units) are given a weight of 10. A logical interface with a small guaranteed rate (CIR) might get an overall weight less than 10. In order to allocate a higher share of the excess bandwidth to logical interfaces with a small guaranteed rate in comparison to the logical interfaces with only shaping rates configured, a minimum weight of 20 is given to the logical interfaces with guaranteed rates configured.

For example, consider a logical interface configuration with five units, as shown in Table 131 on page 918.

Table 131: Allocating Weights with PIR and CIR on Logical Interfaces

Logical Interface (Unit)	Traffic Control Profile	WRR Percentages	Weights
Unit 1	PIR 100 Mbps	95, 0, 0, 5	10, 1, 1, 1
Unit 2	CIR 20 Mbps	25, 25, 25, 25	64, 64, 64, 64
Unit 3	PIR 40 Mbps, CIR 20 Mbps	50, 30, 15, 5	128, 76, 38, 13
Unit 4	Unshaped	95, 0, 0, 5	10, 1, 1, 1
Unit 5	CIR 1 Mbps	95, 0, 0, 5	10, 1, 1, 1

The weights for these units are calculated as follows:

- Select the excess bandwidth sharing proportional rate to be the maximum CIR among all the logical interfaces: 20 Mbps (unit 2).
- Unit 1 has a PIR and unit 4 is unshaped. The weight for these units is 10.
- The weight for unit 1 queue 0 is 9.5 (10 x 95%), which translates to a hardware weight of 10.
- The weight for unit 1 queue 1 is 0 (0 x 0%), but although the weight is zero, a weight of 1 is assigned to give minimal bandwidth to queues with zero WRR.
- Unit 5 has a very small CIR (1 Mbps), and a weight of 20 is assigned to units with a small CIR.
- The weight for unit 5 queue 0 is 19 (20 x 95%), which translates to a hardware weight of 18.
- Unit 3 has a CIR of 20 Mbps, which is the same as the excess bandwidth sharing proportional rate, so it has a total weight of 255.
- The weight of unit 3 queue 0 is 127.5 (255 x 50%), which translates to a hardware weight of 128.

Copyright © 2019, Juniper Networks, Inc.

Sharing Bandwidth Among Logical Interfaces

As a simple example showing how bandwidth is shared among the logical interfaces, assume that all traffic is sent on queue 0. Assume also that there is a 40-Mbps load on all of the logical interfaces. Configuration details are shown in Table 132 on page 919.

NOTE: On the MX960 router, bandwidth sharing across high priority and strict-high priority schedulers configured on logical interfaces might not be as expected. This is a hardware limitation.

Table 132: Sharing Bandwidth Among Logical Interfaces

Logical Interface (Unit)	Traffic Control Profile	WRR Percentages	Weights
Unit 1	PIR 100 Mbps	95, 0, 0, 5	10, 1, 1, 1
Unit 2	CIR 20 Mbps	25, 25, 25, 25	64, 64, 64, 64
Unit 3	PIR 40 Mbps, CIR 20 Mbps	50, 30, 15, 5	128, 76, 38, 13
Unit 4	Unshaped	95, 0, 0, 5	10, 1, 1, 1

1. When the port is shaped at 40 Mbps, because units 2 and 3 have a guaranteed rate (CIR) configured, both units 2 and 3 get 20 Mbps of shared bandwidth.
2. When the port is shaped at 100 Mbps, because units 2 and 3 have a guaranteed rate (CIR) configured, each of them can transmit 20 Mbps. On units 1, 2, 3, and 4, the 60 Mbps of excess bandwidth is shaped according to the values shown in Table 133 on page 919.

Table 133: First Example of Bandwidth Sharing

Logical Interface (Unit)	Calculation	Bandwidth
Unit 1	10 / (10+64+128+10) x 60 Mbps	2.83 Mbps
Unit 2	64 / (10+64+128+10) x 60 Mbps	18.11 Mbps
Unit 3	128 / (10+64+128+10) x 60 Mbps	36.22 Mbps
Unit 4	10 (10+64+128+10) x 60 Mbps	2.83 Mbps

However, unit 3 only has 20 Mbps extra (PIR and CIR) configured. This means that the leftover bandwidth of 16.22 Mbps (36.22 Mbps – 20 Mbps) is shared among units 1, 2, and 4. This is shown in Table 134 on page 920.

Copyright © 2019, Juniper Networks, Inc.

Table 134: Second Example of Bandwidth Sharing

Logical Interface (Unit)	Calculation	Bandwidth
Unit 1	10 / (10+64+128+10) x 16.22 Mbps	1.93 Mbps
Unit 2	64 / (10+64+128+10) x 16.22 Mbps	12.36 Mbps
Unit 4	10 (10+64+128+10) x 16.22 Mbps	1.93 Mbps

Finally, Table 135 on page 920 shows the resulting allocation of bandwidth among the logical interfaces when the port is configured with a 100-Mbps shaping rate.

Table 135: Final Example of Bandwidth Sharing

Logical Interface (Unit)	Calculation	Bandwidth
Unit 1	2.83 Mbps + 1.93 Mbps	4.76 Mbps
Unit 2	20 Mbps + 18.11 Mbps + 12.36 Mbps	50.47 Mbps
Unit 3	20 Mbps + 20 Mbps	40 Mbps
Unit 4	2.83 Mbps + 1.93 Mbps	4.76 Mbps

Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs

You can configure ingress CoS parameters, including hierarchical schedulers, on MX Series routers with Enhanced Queuing DPCs (that is, line cards that have a queuing chip). In general, the supported configuration statements apply to per-unit schedulers or to hierarchical schedulers.

NOTE: Ingress CoS is not supported on line cards that do not contain a queuing chip.

To configure ingress CoS for per-unit schedulers, include the following statements at the **[edit class-of-service interfaces *interface-name*]** hierarchy level:

NOTE: The **input-scheduler-map** and **input-traffic-control-profile** statements are mutually exclusive at the same hierarchy level.

```
[edit class-of-service interfaces interface-name]
input-excess-bandwidth-share (proportional value | equal);
input-scheduler-map map-name;
input-shaping-rate rate;
input-traffic-control-profile profile-name shared-instance instance-name;
unit logical-unit-number;
```


Copyright © 2019, Juniper Networks, Inc.

```
    input-scheduler-map map-name;
    input-shaping-rate (percent percentage | rate);
    input-traffic-control-profile profile-name shared-instance instance-name;
}
```

To configure ingress CoS for hierarchical schedulers, include the **interface-set** *interface-set-name* statement at the **[edit class-of-service interfaces]** hierarchy level:

```
[edit class-of-service interfaces]
interface-set interface-set-name {
    input-excess-bandwidth-share (proportional value | equal);
    input-traffic-control-profile profile-name shared-instance instance-name;
    input-traffic-control-profile-remaining profile-name;
    interface interface-name {
        input-excess-bandwidth-share (proportional value | equal);
        input-traffic-control-profile profile-name shared-instance instance-name;
        input-traffic-control-profile-remaining profile-name;
        unit logical-unit-number;
            input-traffic-control-profile profile-name shared-instance instance-name;
        }
    }
}
```

By default, ingress CoS features are disabled on the Enhanced Queuing DPC.

For an Enhanced Queuing (EQ) DPC on an MX Series router, CoS queuing and scheduling are enabled on the egress side but disabled on the ingress side by default. To enable ingress CoS on the EQ DPC, you must configure the **traffic-manager** statement with **ingress-and-egress** mode:

```
[edit chassis fpc slot-number pic pic-number]
traffic-manager mode ingress-and-egress;
```

NOTE: If you enable ingress CoS settings and inline services on the same FPC, the FPC moves to the offline state. This behavior is expected because traffic black hole detection is triggered that causes the FPC to move to the offline state.

Configured CoS features on the ingress are independent of CoS features on the egress, with the following exceptions:

- If you configure a per-unit or hierarchical scheduler at the **[edit class-of-service interfaces]** hierarchy level, the schedulers apply in both the ingress and egress directions.
- You cannot configure the same logical interface on an ingress and an egress interface set. A logical interface can only belong to one interface set.
- The DPC's frame buffer of 512 MB is shared between ingress and egress configurations.

Copyright © 2019, Juniper Networks, Inc.

The following behavior aggregate (BA) classification tables are supported on the ingress side of the Enhanced Queuing DPC:

- inet-precedence
- DSCP
- exp (MPLS)
- DSCP for IPv6
- IEEE 802.1p

Related Documentation

- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- Enhanced Queuing DPC CoS Properties on page 908

Configuring Customer VLAN (Level 3) Shaping on Enhanced Queuing DPCs

Customer VLAN (level 3) shaping on an MX Series 10-Gigabit Ethernet Enhanced Queuing DPC differs from the customer VLAN (level 3) shaping on an MX Series 1-Gigabit Ethernet Enhanced Queuing DPC. To use the customer VLAN (level 3) shaping on an MX Series 10-Gigabit Ethernet Enhanced Queuing DPC, configure an interface set at the **[edit interfaces interface-set]** hierarchy level. You do not need to configure the interface set while using customer VLAN (level 3) on an MX Series 1-Gigabit Ethernet Enhanced Queuing DPC.

To configure customer VLAN (level 3) shaping on an MX Series 10-Gigabit Ethernet Enhanced Queuing DPC:

1. Configure the interface set at the **[edit interfaces]** hierarchy level.

```
[edit interfaces]
user@host# set interface-set jnpr interface unit 100
user@host# set interface-set jnpr interface xe-1/0/0 unit 101
```

2. Configure the hierarchical scheduler and enable VLAN tagging.

```
[edit interfaces]
user@host# set xe-1/0/0 hierarchical-scheduler
user@host# set xe-1/0/0 vlan-tagging
```

3. Configure the logical interface properties.

```
[edit interfaces]
user@host# set xe-1/0/0 unit 100 vlan-id 100
user@host# set xe-1/0/0 unit 100 family inet address 10.1.0.1/24
user@host# set xe-1/0/0 unit 101 vlan-id 101
user@host# set xe-1/0/0 unit 101 family inet address 10.1.1.1/24
```


Copyright © 2019, Juniper Networks, Inc.

4. Configure the traffic control profiles at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
user@host# set traffic-control-profiles profile1 shaping-rate 10g burst-size 2k
user@host# set traffic-control-profiles profile1 guaranteed-rate 10g burst-size 2k
user@host# set traffic-control-profiles profile2 shaping-rate 50m burst-size 2k
user@host# set traffic-control-profiles profile2 guaranteed-rate 50m burst-size 2k
user@host# set traffic-control-profiles profile3 shaping-rate 80m burst-size 3k
user@host# set traffic-control-profiles profile3 guaranteed-rate 80m burst-size 3k
```

5. Configure the output traffic control profiles at the **[edit class-of-service interfaces]** hierarchy level.

```
[edit class-of-service interfaces]
user@host# set interface-set jnpr output-traffic-control-profiles profile1
user@host# set xe-1/0/0 unit 100 output-traffic-control-profiles profile2
user@host# set xe-1/0/0 unit 101 output-traffic-control-profiles profile3
```

To configure customer VLAN (level 3) shaping on an MX Series 1-Gigabit Ethernet Enhanced Queuing DPC:

1. Configure the interface set at the **[edit interfaces]** hierarchy level.

```
[edit interfaces]
user@host# set interface ge-1/0/0 unit 100
user@host# set interface ge-1/0/0 unit 101
```

2. Configure the hierarchical scheduler and enable the VLAN tagging.

```
[edit interfaces]
user@host# set ge-1/0/0 hierarchical-scheduler
user@host# set ge-1/0/0 vlan-tagging
```

3. Configure the logical interface properties.

```
[edit interfaces]
user@host# set ge-1/0/0 unit 100 vlan-id 100
user@host# set ge-1/0/0 unit 100 family inet address 10.1.0.1/24
user@host# set ge-1/0/0 unit 101 vlan-id 101
user@host# set ge-1/0/0 unit 101 family inet address 10.1.1.1/24
```

4. Configure the traffic control profiles at the **[edit class-of-service]** hierarchy level.

```
[edit class-of-service]
user@host# set traffic-control-profiles profile1 shaping-rate 10g burst-size 2k
user@host# set traffic-control-profiles profile1 guaranteed-rate 10g burst-size 2k
user@host# set traffic-control-profiles profile2 shaping-rate 50m burst-size 2k
user@host# set traffic-control-profiles profile2 guaranteed-rate 50m burst-size 2k
user@host# set traffic-control-profiles profile3 shaping-rate 80m burst-size 3k
```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set traffic-control-profiles profile3 guaranteed-rate 80m burst-size 3k
```

5. Configure the traffic control profiles at the **[edit class-of-service interfaces]** hierarchy level.

```
[edit class-of-service interfaces]
user@host# set ge-1/0/0 unit 100 output-traffic-control-profiles profile2
user@host# set ge-1/0/0 unit 101 output-traffic-control-profiles profile3
```

Related Documentation

- Enhanced Queuing DPC CoS Properties on page 908

Simple Filters Overview

Simple filters are recommended for metropolitan Ethernet applications. They are supported on Gigabit Ethernet intelligent queuing 2 (IQ2) and Enhanced Queuing Dense Port Concentrator (DPC) interfaces only.

Unlike normal filters, simple filters are for IPv4 traffic only and have the following restrictions:

- The **next term** action is not supported.
- Qualifiers, such as the **except** and **protocol-except** statements, are not supported.
- Noncontiguous masks are not supported.
- Multiple source addresses and destination addresses in a single term are not supported. If you configure multiple addresses, only the last one is used.
- Ranges are only valid as source or destination ports. For example, **source-port 400-500** or **destination-port 600-700**.
- Output filters are not supported. You can apply a simple filter to ingress traffic only.
- Simple filters are not supported for interfaces in an aggregated-Ethernet bundle.
- Explicitly configurable terminating actions, such as **accept**, **reject**, and **discard**, are not supported. Simple filters always accept packets.

NOTE: On the MX Series routers with the Enhanced Queuing DPC, the forwarding class is not supported as a from match condition.

Related Documentation

- Configuring a Simple Filter on page 839

Copyright © 2019, Juniper Networks, Inc.

Configuring Simple Filters on Enhanced Queuing DPCs

You can configure and apply a simple filter to perform multifield classification on the ingress interfaces of an MX Series router with Enhanced Queuing DPCs. These simple filters can be used to override default CoS classification parameters such as forwarding class or loss priority. Simple filters, in contrast to other firewall filters, only support a subset of the full firewall filter syntax.

To configure a simple filter, include the **simple-filter** statement at the **[edit firewall family inet]** hierarchy level:

```
[edit firewall family inet]
simple-filter filter-name {
  term term-name {
    from {
      ... match-conditions...
    }
    then {
      forwarding-class class-name;
      loss-priority priority;
    }
  }
}
```

The following example configures a simple filter to detect ingress packets from various source addresses (**10.1.1.1/32**, **10.10.10.10/32**, and **10.4.0.0/8**), destination addresses (**10.6.6.6/32**), protocols (**tcp**), and source ports (**400-500**, **http**). The filter then assigns various forwarding classes and loss priorities to the filtered traffic. Finally, the filter is applied to the input side of an Enhanced Queuing DPC interface (**ge-2/3/3**).

```
[edit]
firewall {
  family inet {
    simple-filter sf-for-eq-dpc {
      term 1 {
        from {
          source-address 10.1.1.1/32;
          protocol tcp;
        }
        then loss-priority low;
      }
      term 2 {
        from {
          source-address 10.4.0.0/8;
          source-port http;
        }
        then loss-priority high;
      }
      term 3 {
        from {
          destination-address 10.6.6.6/32;
          source-port 400-500;
```

Copyright © 2019, Juniper Networks, Inc.

```
            }
            then {
              loss-priority low;
              forwarding-class best-effort;
            }
          }
          term 4 {
            from {
              forwarding-class expedited-forwarding;
              source-address 10.10.10.10/32;
            }
            then loss-priority low;
          }
          term 5 {
            from {
              source-address 10.10.10.10/32;
            }
            then loss-priority low;
          }
        }
      }
    }
    interfaces { # Apply the simple filter above to the input side of the interface.
    ge-2/3/3 {
      unit 0 {
        family inet {
          simple-filter {
            input sf-for-eq-dpc;
          }
        }
      }
    }
```

Related Documentation

- *Simple Filter Overview*
- *How Simple Filters Evaluate Packets*
- *Guidelines for Configuring Simple Filters*
- *Guidelines for Applying Simple Filters*

Configuring a Simple Filter

This simple filter sets the loss priority to low for TCP traffic with source address **10.1.1.1**, sets the loss priority to high for HTTP (web) traffic with source addresses in the **203.0.113.0/24** range, and sets the loss priority to low for all traffic with destination address **10.6.6.6**. The simple filter is applied as an input filter (arriving packets are checking for destination address **10.6.6.6**, not queued output packets) on interface **ge-0/0/1.0**.

```
[edit]
firewall {
  family inet {
```


Copyright © 2019, Juniper Networks, Inc.

```
    simple-filter filter1 {
      term 1 {
        from {
          source-address {
            10.1.1.1/32;
          }
          protocol {
            tcp;
          }
        }
        then loss-priority low;
      }
      term 2 {
        from {
          source-address {
            203.0.113.0/24;
          }
          source-port {
            http;
          }
        }
        then loss-priority high;
      }
      term 3 {
        from {
          destination-address {
            10.6.6.6/32;
          }
        }
        then {
          loss-priority low;
          forwarding-class best-effort;
        }
      }
    }
  }
}
interfaces {
  ge-0/0/1 {
    unit 0 {
      family inet {
        simple-filter {
          input filter1;
        }
        address 10.1.2.3/30;
      }
    }
  }
}
```

Related Documentation

- Simple Filters Overview on page 839

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 26

Configuring Class of Service on MICs, MPCs, and MLCs

- CoS Features and Limitations on MIC and MPC Interfaces on page 930
- Dedicated Queue Scaling for CoS Configurations on MIC and MPC Interfaces Overview on page 931
- Verifying the Number of Dedicated Queues Configured on MIC and MPC Interfaces on page 934
- Scaling of Per-VLAN Queuing on Non-Queuing MPCs on page 935
- Increasing Available Bandwidth on Rich-Queuing MPCs by Bypassing the Queuing Chip on page 940
- Flexible Queuing Mode on page 941
- Multifield Classifier for Ingress Queuing on MX Series Routers with MPC on page 944
- Example: Configuring a Filter for Use as an Ingress Queuing Filter on page 945
- Ingress Queuing Filter with Policing Functionality on page 948
- Ingress Rate Limiting on MX Series Routers with MPCs on page 953
- Rate Shaping on MIC and MPC Interfaces on page 954
- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- Example: Configuring Per-Priority Shaping on MIC and MPC Interfaces on page 960
- Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates on page 966
- Example: Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates on page 967
- Traffic Burst Management on MIC and MPC Interfaces Overview on page 970
- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972
- Configuring Ingress Hierarchical CoS on MIC and MPC Interfaces on page 975
- Configuring a CoS Scheduling Policy on Logical Tunnel Interfaces on page 977
- Per-Unit Queuing and Hierarchical Queuing for MIC and MPC Interfaces on page 978
- Managing Dedicated and Remaining Queues for Static CoS Configurations on MIC and MPC Interfaces on page 981
- Excess Bandwidth Distribution on MIC and MPC Interfaces Overview on page 983

Copyright © 2019, Juniper Networks, Inc.

- Bandwidth Management for Downstream Traffic in Edge Networks Overview on page 984
- Scheduler Delay Buffering on MIC and MPC Interfaces on page 986
- Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 986
- Drop Profiles on MIC and MPC Interfaces on page 989
- Intelligent Oversubscription on MIC and MPC Interfaces Overview on page 990
- Jitter Reduction in Hierarchical CoS Queues on page 991
- Example: Reducing Jitter in Hierarchical CoS Queues on page 993
- CoS on Ethernet Pseudowires in Universal Edge Networks Overview on page 999
- CoS Scheduling Policy on Logical Tunnel Interfaces Overview on page 1000
- Configuring CoS on an Ethernet Pseudowire for Multiservice Edge Networks on page 1000
- CoS for L2TP LNS Inline Services Overview on page 1001
- Configuring Static CoS for an L2TP LNS Inline Service on page 1003
- CoS on Circuit Emulation ATM MICs Overview on page 1005
- Configuring CoS on Circuit Emulation ATM MICs on page 1006
- Understanding IEEE 802.1p Inheritance push and swap from a Transparent Tag on page 1008
- Configuring IEEE 802.1p Inheritance push and swap from the Transparent Tag on page 1009
- CoS on Application Services Modular Line Card Overview on page 1011

CoS Features and Limitations on MIC and MPC Interfaces

MIC and MPC interfaces on MX Series 5G Universal Routing Platforms use the Trio chipset-based queuing model, which supports CoS characteristics that are optimized compared to CoS characteristics supported by the standard queuing model. However, some CoS features are not supported or are supported with limitations on MIC and MPC interfaces.

When configuring CoS features on MIC and MPC interfaces on MX Series routers, keep the following limitations in mind.

Table 136: CoS Limitations on MIC and MPC Interfaces

CoS Feature	Limitation on MIC or MPC Interfaces
Classifiers	Interfaces on MPCs support up to 32 classifiers of each type per module.
BA classifier for MPLS packets	When you configure a behavior aggregate (BA) classifier that does not include a specific rewrite rule for MPLS packets, we highly recommend that you enable the default MPLS EXP classifier. Doing so ensures that MPLS exp value is rewritten according to the BA classifier rules configured for forwarding or packet loss priority. For more information, see "Default MPLS EXP Classifier" on page 42. To enable the default MPLS EXP classifier, include the **default** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules exp]** hierarchy level.

Copyright © 2019, Juniper Networks, Inc.

Table 136: CoS Limitations on MIC and MPC Interfaces (continued)

CoS Feature	Limitation on MIC or MPC Interfaces
Rewrite rules	For interfaces on MPCs or on MICs installed in MPCs, you can figure up to 32 rewrite rules: • DSCP rewrite rules • Internet rewrite rules • EXP rewrite rules • IEEE rewrite rules However, if you configure all 32 allowed rewrite rules, the class-of-service process can intermittently fail and generate syslog entries.
Default rewrite rules for MPLS-enabled interfaces	On interfaces other than MIC and MPC interfaces, the default EXP rewrite rule is automatically applied to MPLS-enabled interfaces, even if not configured. On MIC and MPC interfaces, you must explicitly configure EXP rewrite rules to MPLS-enabled interfaces.
Rewrite rules for service VLAN tag CoS bits	For MIC and MPC interfaces for VPLS or bridge domains, rewrite service VLAN tag CoS bits by configuring the rewrite rules on the *core-facing* interface.
Excess bandwidth sharing	Interfaces on MICs and MPCs do not support the **excess-bandwidth-share** configuration statement, which specifies how excess bandwidth at an interface set in a hierarchical scheduler environment is to be shared: proportionally or equally. Instead, you can include the excess-rate statement at one of the following hierarchy levels: • **[edit class-of-service** schedulers ***scheduler-name*****]** • **[edit class-of-service** traffic-control-profiles ***traffic-control-profile-name*****]**
Layer 1 and Layer 2 overhead	MIC and MPC interfaces take all Layer 1 and Layer 2 overhead bytes into account for all levels of the hierarchy, including preamble, interpacket gaps, frame check sequence, and cyclical redundancy check. Queue statistics also take these overheads into account when displaying byte statistics.
Pairing of load-balancing links	When load balancing EQ MIC interfaces installed in Type 1 MPCs, you should configure odd- and even-numbered interfaces in the form ***interface-fpc/odd | even/*ports**. For example, if one link is **xe-1/0/0**, the other should be **xe-1/1/0**. If you do not configure odd and even load balancing, the system RED-drops packets when sending at line rate. This limitation does not apply to interfaces on EQ MICs installed in Type 2 MPCs.

Related Documentation

- Rate Shaping on MIC and MPC Interfaces on page 954
- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972
- Scheduler Delay Buffering on MIC and MPC Interfaces on page 986
- Drop Profiles on MIC and MPC Interfaces on page 989

Dedicated Queue Scaling for CoS Configurations on MIC and MPC Interfaces Overview

Queuing Ethernet Modular Port Concentrators (MPCs) provide a set of dedicated queues for subscriber interfaces configured with hierarchical scheduling or per-unit scheduling.

Copyright © 2019, Juniper Networks, Inc.

The dedicated queues offered on these MPCs enable service providers to reduce costs through different scaling configurations. These queuing MPCs enable service providers to reduce the cost per subscriber by allowing many subscriber interfaces to be created with four or eight queues.

This topic describes the overall queue, scheduler node, and logical interface scaling for subscriber interfaces created on these MIC and MPC combinations.

Queue Scaling for MPCs

Beginning with Junos OS Release 15.1, MPC2E-3D-NG-Q, MPC3E-3D-NG-Q, MPC5EQ-40G10G, and MPC5EQ-100G10G MPCs support up to five levels of hierarchical queuing. Beginning with Junos OS Release 16.1R1, MPC7 line cards also support five levels of hierarchical queuing. Table 137 on page 932 lists the number of dedicated queues and nodes supported per MPC.

Table 137: Dedicated Queues for MPCs

MPC	Dedicated Queues	Level 4 Nodes	Level 3 Nodes	Level 2 Nodes	Level 1 Nodes (Ports)
MPC2E-3D-NG-Q MPC3E-3D-NG-Q	512,000	64,000	16,000	4000	384
MPC5EQ-40G10G MPC5EQ-100G10G	1 million	128,000	32,000	4000	384
MPC7	256,000	32,000	8000	4000	126

CAUTION: The maximum scaling targets provided in Table 137 on page 932 are based on system level design specifications. Actual realized subscriber or session scale is highly dependent upon the configuration and can be influenced by configuration variables including: the number of routes, the number of enabled services, the number of policy and firewall filters, policers, counters, statistics and access model type. Once you define a configuration, your Juniper account team can help characterize the expected system level scale or scale range for your live deployment.

MPCs vary in the number of Packet Forwarding Engines on board. MPC2E-3D-NG-Q and MPC3E-3D-NG-Q MPCs each have one Packet Forwarding Engine, allowing all 64,000 level 4 (subscriber) nodes to be allocated to a single MIC. MPC5EQ MPCs have two Packet Forwarding Engines, one for each possible MIC, each supporting 64,000 level 4 (subscriber) nodes.

Copyright © 2019, Juniper Networks, Inc.

NOTE: The nonqueuing MPCs MPC2E-3D-NG, MPC3E-3D-NG, MPC5E-40G10G, and MPC5E-100G10G provide up to eight queues per port in standard configuration. However, each of these MPCs can be configured to provide limited-scale hierarchical class of service (HCoS) and up to 32,000 queues.

Managing Remaining Queues

In Junos OS releases earlier than Release 15.1R4, SNMP traps generate system log messages to notify you:

- When the number of available dedicated queues on the MPC drops below 10 percent. For example:

```
Mar 15 14:55:22.977 host cosd[1963]: COSD_OUT_OF_DEDICATED_QUEUES: Queue usage
 count for interface xe-3/0/0 is at 90 percent
```

- When the maximum number of dedicated queues on the MPCs is reached. For example,

```
Mar 15 18:01:59.344 host cosd[3848]: COSD_OUT_OF_DEDICATED_QUEUES: Queue usage
 count for interface xe-3/0/0 is at 100 percent.
```

  When the maximum number of dedicated queues is allocated, the system does not provide subsequent subscriber interfaces with a dedicated set of queues. For per-unit scheduling configurations, there are no configurable queues remaining on the MPC.

For hierarchical scheduling configurations, remaining queues are available when the maximum number of dedicated queues is reached on the MPC. Traffic from these logical interfaces is considered unclassified and attached to a common set of queues that are shared by all subsequent logical interfaces. These common queues are the default port queues that are created for every port. You can configure a traffic-control profile and attach that to the interface to provide CoS parameters for the remaining queues. These subscriber interfaces remain with this traffic-control profile, even if dedicated queues become available.

NOTE: Starting in Junos OS Release 15.1R4, the COSD_OUT_OF_DEDICATED_QUEUES functionality is not available for QoS-enabled dynamic subscribers. Starting in Junos OS Release 17.4R1, CoS resource monitoring enables you to set a per-FPC queue threshold of up to 90 percent of resources bound to a scheduling hierarchy; subscriber logins are not allowed when the threshold is reached. However, this threshold applies to all queues, not dedicated queues alone. See *Resource Monitoring for Subscriber Management and Services Overview* for more information.

Copyright © 2019, Juniper Networks, Inc.

Release History Table

Release	Description
16.1R1	Beginning with Junos OS Release 16.1R1, MPC7 line cards also support five levels of hierarchical queuing.
15.1R1	Beginning with Junos OS Release 15.1, MPC2E-3D-NG-Q, MPC3E-3D-NG-Q, MPC5EQ-40G10G, and MPC5EQ-100G10G MPCs support up to five levels of hierarchical queuing.

Related Documentation

- *Hierarchical Class of Service Feature Guide*
- Understanding Hierarchical Scheduling on page 337
- Managing Dedicated and Remaining Queues for Static CoS Configurations on MIC and MPC Interfaces on page 981
- *Managing Dedicated and Remaining Queues for Dynamic CoS Configurations on MIC and MPC Interfaces*
- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972

Verifying the Number of Dedicated Queues Configured on MIC and MPC Interfaces

Purpose Display the number of dedicated queue resources that are configured for the logical interfaces on a port.

Copyright © 2019, Juniper Networks, Inc.

Action

```
user@host#show class-of-service interface ge-1/1/0
Physical interface: ge-1/1/0, Index: 166
Queues supported: 4, Queues in use: 4
Total non-default queues created: 4
  Scheduler map: <default>, Index: 2
  Chassis scheduler map: <default-chassis>, Index: 4

  Logical interface: ge-1/1/0.100, Index: 72, Dedicated Queues: no
    Shaping rate: 32000
    Object              Name                    Type                  Index
    Scheduler-map       <remaining>                                   0
    Classifier          ipprec-compatibility    ip                    13

  Logical interface: ge-1/1/0.101, Index: 73, Dedicated Queues: no
    Shaping rate: 32000
    Object              Name                    Type                  Index
    Scheduler-map       <remaining>                                   0
    Classifier          ipprec-compatibility    ip                    13

  Logical interface: ge-1/1/0.102, Index: 74, Dedicated Queues: yes
    Shaping rate: 32000
    Object                   Name               Type                  Index
    Traffic-control-profile  <control_tc_prof>  Output                45866
```

Related Documentation

- Managing Dedicated and Remaining Queues for Static CoS Configurations on MIC and MPC Interfaces on page 981
- *Managing Dedicated and Remaining Queues for Dynamic CoS Configurations on MIC and MPC Interfaces*

Scaling of Per-VLAN Queuing on Non-Queuing MPCs

Per-VLAN (logical interface) queueing has been introduced on most MPCs supported on the MX platform. Table 138 on page 935 shows the details along with the supported JUNOS release.

Table 138: MPC and MIC support for per-VLAN (logical interface) queuing

MPC	MICs Supported	JUNOS Release
16x10GE MPC	N/A	13.2
MPC3E	2x10GE XFP	13.2
	10x10GE SFPP	13.2
	2x40G QSFPP	13.2
	1x100GE CXP	13.2
	1x100G CFP	13.2

Copyright © 2019, Juniper Networks, Inc.

Table 138: MPC and MIC support for per-VLAN (logical interface) queuing (continued)

MPC	MICs Supported	JUNOS Release
MPC4E-32x10GE SFPP	N/A	13.3
MPC4E-2x100GE+8x10GE SFPP	N/A	13.3
MPC6E	24x10GE SFPP	15.1
	24x10GE SFFP OTN	15.1
	2x100GE CFP2 OTN	15.1
	4x100GE CXP	15.1
MPC5E-10G100G	N/A	13.3R3
MPC5E-10G40G	N/A	13.3R3
MPC2E-3D-NG/MPC3E-3D-NG	20x1GE SFP	15.1
	2xGE-XFP	15.1
	40x1GE	15.1
	4xGE-XFP	15.1
	8OC3OC12-4OC48	15.1
	4OC3OC12-1OC48	15.1
	8CHOC3-4CHOC12	15.1
	4CHOC3-1CHOC12	15.1
	8DS3-E3	15.1
	1xOC192-XFP	15.1
	16xCHE1-T1-CE	15.1
	8OC3-2OC12-ATM-CC-CE	15.1
	4COC3-1COC12-CE	15.1
	20xGE-SFP-E	15.1
MPC3E-3D-NG	2x10GE XFP	15.1
	10x10GE SFPP	15.1

Copyright © 2019, Juniper Networks, Inc.

Table 138: MPC and MIC support for per-VLAN (logical interface) queuing (continued)

MPC	MICs Supported	JUNOS Release
	2x40G QSFPP	15.1
	1x100GE CXP	15.1

To enable logical interface scheduling, you include the **per-unit-scheduler** statement at the **[edit interfaces *interface name*]** hierarchy level. When per-unit schedulers are enabled, you can define dedicated schedulers for logical interfaces by including the **scheduler-map** statement at the **[edit class-of-service interfaces *interface name* unit *logical unit number*]** hierarchy level. Alternatively, you can include the **scheduler-map** statement at the **[edit class-of-service traffic-control-profiles *traffic control profile name*]** hierarchy level and then include the **output-traffic-control-profile** statement at the **[edit class-of-service interfaces *interface name* unit *logical unit number*]** hierarchy level.

Table 139 on page 937 shows the number of VLANs per port available in both 8-queue and 4-queue mode for 16x10GE, MPC3E, MPC4E and MPC6E.

Table 139: Number of VLANs on 16x10G, MPC3E, MPC4E and MPC6E

MPC	MIC	VLANs/Port – 8-Queue Mode	VLANs/Port – 4-Queue Mode
16X10GE	No	21	44
MPC3E	2x10GE with XFP	20	42
	10X10GE with SFP+	12 per group of 5 ports*	34 per group of 5 ports*
	2X40GE with QSFP+	20	42
	1X100GE with CXP	20	42
32x10GE MPC4E	No	20 per group of 4 ports*	48 per group of 4 ports*
2x100GE + 8x10GE MPC4E	No	26	54
MPC6E	24X10GE	20 per group of 3 ports*	42 per group of 3 ports*
	2X100GE with CFP2 OTN	26	54
	4X100GE MIC with CXP	21	44

*The 10X10GE MIC for the MPC3E, the 32X10GE MPC4E, and the 24X10GE MICs for the MPC6E share VLANs across a port group. You can assign all of the available VLANs to one port within the port group or spread them across the ports in any combination.

Copyright © 2019, Juniper Networks, Inc.

Enabling and configuring per-unit schedulers on these interfaces adds additional output to the **show interfaces *interface name* [detail | extensive]** command. This additional output lists the maximum resources available and the number of configured resources for schedulers. Following is sample output showing the CoS scheduler resource information on a non-queuing line card:

```
root@R1# run show interfaces et-2/2/0 detail

Physical interface: et-2/2/0, Enabled, Physical link is Down
  Interface index: 165, SNMP ifIndex: 550, Generation: 168
 Link-level type: Ethernet, MTU: 1522, Speed: 100Gbps, BPDU Error: None, Loopback:
Disabled, Source filtering: Disabled,
 Flow control: Enabled
 Device flags   : Present Running Down
 Interface flags: Hardware-Down SNMP-Traps Internal: 0x4000
 Link flags     : Scheduler
 CoS queues     : 8 supported, 8 maximum usable queues
 Schedulers     : 0
 Hold-times     : Up 0 ms, Down 0 ms
 Current address: 80:71:1f:10:e6:b4, Hardware address: 80:71:1f:10:e6:b4
 Last flapped   : 2013-05-07 16:17:01 PDT (03:16:41 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 transit statistics:
   Input  bytes  :                   0
   Output bytes  :                   0
   Input  packets:                   0
   Output packets:                   0
 Egress queues: 8 supported, 4 in use
 CoS scheduler resource information:
   Maximum units supported per MIC/PIC: 20
   Configured units per MIC/PIC: 1
   Maximum units allowed per port: 20
   Configured units on this port: 1
 Queue counters:       Queued packets  Transmitted packets      Dropped packets

   0 best-effort                    0                    0                    0

   1 expedited-fo                   0                    0                    0

   2 assured-forw                   0                    0                    0

   3 network-cont                   0                    0                    0

 Queue number:         Mapped forwarding classes
   0                   best-effort
```

If you enable more VLANs than the previously mentioned MPC/MIC combinations support, VLANs up to the supported numbers receive dedicated queuing resources. The additional VLANs share port queues. Scheduling for port queues cannot be controlled. However, port queues are guaranteed 1 percent of the physical interface bandwidth to avoid queue starving and buffer holdup.

Copyright © 2019, Juniper Networks, Inc.

In the case of MPC2E-NG/3E-NG, MPC5E and MPC7E/8E/9E SKUs, the following command needs to be configured to enable "flexible queuing" on the MPC. Configuration of this knob results in a reboot of the MPC. The per-unit-scheduler, hierarchical scheduling and 2 level hierarchical scheduling are supported. There are 32K queues enabled and they can be used for either ingress queueing or egress queueing. The 32K queues are available when all 8 queues are used per IFL.

```
chassis {
  fpc 0 {
    flexible-queuing-mode; }
      }
```

Table 140 on page 939 shows the number of VLANs per port available in both 8-queue and 4-queue mode for MPC3E-NG/MPC2E-NG, and MPC5E.

Table 140: Number of VLANs on MPC3E-NG/MPC2E-NG, MPC5E

MPC	MIC	VLANs/Port – 8-Queue Mode	VLANs/Port – 4-Queue Mode
MPC3E-NG/MPC2E-NG	Supported MICs	32K	32K
MPC5E	Supported MICs	32K	32K

NOTE: The number of logical interfaces with per-vlan queuing enabled should not exceed line card maximum. If the line card maximum is exceeded, then the queuing behavior is unpredictable. This could mean that some logical interfaces have queues assigned and some do not.

Related Documentation

- per-unit-scheduler on page 1226
- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300

Copyright © 2019, Juniper Networks, Inc.

Increasing Available Bandwidth on Rich-Queuing MPCs by Bypassing the Queuing Chip

Queuing MPCs contain a queuing chip that enables rich-queuing features such as hierarchical and per-vlan queuing. By default, all traffic passing through an interface on one of these MPCs also passes through the queuing chip, which decreases the available bandwidth of the interface. If you do not require hierarchical or per-vlan queuing on a particular interface of a queuing MPC, you can bypass the queuing chip to increase the available bandwidth.

Starting with Junos OS 18.2R1, you can enable this option on vMX routers to save a vCPU when scheduling is not needed on an interface.

To bypass the queuing chip on a queuing MPC, you must be running Junos OS Release 14.2 or later. On vMX routers, you must be running Junos OS Release 18.2 or later. You can bypass the queuing chip on the following line cards:

- *MPC1 Q*
- *MPC1E Q*
- *MPC2 Q*
- *MPC2 EQ*
- *MPC2E Q*
- *MPC2E EQ*
- MPC5E Q (*2x100GE + 4x10GE MPC5EQ* or *6x40GE + 24x10GE MPC5EQ*)

To bypass the queuing chip on an interface on a queuing MPC:

1. Ensure that neither per-unit-scheduler nor hierarchical-scheduler is configured on the interface.

   NOTE: It is not possible to bypass the queuing chip on an interface if per-unit or hierarchical scheduling is configured on that interface.

2. Enable bypass-queuing-chip on the interface.

   For example:

   ```
   [edit interfaces]
   user@router# set interface- name bypass-queuing-chip
   ```

3. Commit your changes.

   ```
   [edit interfaces]
   user@router# show
   interface-name {
   ```


Copyright © 2019, Juniper Networks, Inc.

```
    bypass-queueing-chip;
}
```

4. Verify your changes.

```
user@router> show interfaces interface-name
Physical interface: interface-name, Enabled, Physical link is Up
  Interface index: 147, SNMP ifIndex: 524
  Link-level type: Ethernet, MTU: 1514, MRU: 1522, LAN-PHY mode, Speed:
1000mbps,
  BPDU Error: None, MAC-REWRITE Error: None, Loopback: Disabled, Source
filtering: Disabled,
  Flow control: Enabled, Auto-negotiation: Enabled, Remote fault: Online
  Pad to minimum frame size: Disabled
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  Link flags     : None
  CoS queues     : 8 supported, 4 maximum usable queues
  Schedulers     : 0, Queuing Chip Bypassed
  Current address: 00:21:59:0f:35:31, Hardware address: 00:21:59:0f:35:31
  Last flapped   : 2014-04-29 14:10:18 PDT (02:27:46 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : None
  Active defects : None
  Interface transmit statistics: Disabled
```

Release History Table

Release	Description
18.2R1	Starting with Junos OS 18.2R1, you can enable this option on vMX routers to save a vCPU when scheduling is not needed on an interface.
14.2	To bypass the queuing chip on a queuing MPC, you must be running Junos OS Release 14.2 or later. On vMX routers, you must be running Junos OS Release 18.2 or later.

Related Documentation

- bypass-queuing-chip on page 1058

Flexible Queuing Mode

You can configure the non-hierarchical quality-of-service (non-HQoS) MPCs to support port-based flexible queuing. By default, the non-HQoS MPCs do not support queuing. To enable queuing, you must upgrade these MPCs through an add-on license. After you upgrade these MPCs, they can support a flexible queuing capability of up to 32,000 queues per port and per card, including queues on both ingress and egress interfaces.

Copyright © 2019, Juniper Networks, Inc.

Channelized MICs are supported on non-HQoS MPCs only when flexible queuing is configured.

- Flexible Queuing Mode Overview on page 942
- Upgrading non-HQoS MPCs to Support Flexible Queuing on page 943
- Disabling Flexible Queuing for non-HQoS MPCs to Optimize Power Utilization on page 943

Flexible Queuing Mode Overview

The queuing component on non-HQoS MPCs is disabled by default to save power. When flexible queuing is enabled on a non-HQoS MPC, the MPC is restarted with the queuing component enabled. The MPC is powered on only if the PEM has sufficient power to bring up the MPC with the queuing component enabled. The MPC remains offline if the required power is not available.

You can enable flexible queuing on the non-HQoS MPCs by including the flexible-queuing-mode statement at the **[edit chassis fpc]** hierarchy level. When queuing is configured, the power consumed by the queuing components at the configured ambient temperature is considered when power is allocated for the MPC.

NOTE: The following MICs are supported on non-HQoS MPCs only when flexible queuing is enabled:

- MIC-3D-8CHOC3-4CHOC12
- MIC-3D-4CHOC3-2CHOC12
- MIC-4COC3-2COC12-G
- MIC-2COC3-1COC12-G

Table 141 on page 942 lists the MPCs that support flexible queueing and the supported Junos OS release for these MPCs.

Table 141: MPCs and the Junos OS Release that Support Flexible Queuing

MPCs	First Supported Junos OS Release
MPC2E-3D-NG	15.1R1
MPC3E-3D-NG	15.1R1
MPC5E	14.1R1
MPC7E-MRATE	15.1F4
MPC7E-10G	15.1F5
MPC8E	
MPC9E	

Copyright © 2019, Juniper Networks, Inc.

Upgrading non-HQoS MPCs to Support Flexible Queuing

You can enable flexible queuing on a non-HQoS MPC to support a maximum of up to 32,000 queues per port and per card, including queues on both ingress and egress interfaces.

This topic describes how to enable flexible queuing on a non-HQoS MPC.

To configure flexible queuing on non-HQoS MPCs:

1. Run the **set chassis fpc *slot-number* flexible-queuing-mode** configuration mode command.

   For example, to configure flexible queuing on an MPC in slot 2:

   ```
   [edit]
   user@router# set chassis fpc 2 flexible-queuing-mode
   ```

   NOTE: When flexible queuing is enabled, the MPC is restarted with the queuing component enabled. The MPC comes online only if the power entry module (PEM) has sufficient power to bring up the MPC with the queuing component enabled. The MPC remains offline if the required power is not available in the PEM.

2. Review your configuration and issue the **commit** command.

   ```
   [edit]
   user@router# commit
   [edit]
     'chassis fpc'
       WARNING: FPC configuration for flexible-queuing is changed. FPC would
   undergo reboot to enable flexible-queuing. FPC would come online only if power
    available is sufficient to enable queuing components.
   commit complete
   ```

Disabling Flexible Queuing for non-HQoS MPCs to Optimize Power Utilization

You can optimize power utilization by disabling flexible queuing on a non-HQoS MPC.

This topic describes how to disable flexible queuing on a non-HQoS MPC.

1. Run the **delete chassis fpc *slot-number* flexible-queuing-mode** command at the **[edit chassis]** hierarchy level.

   For example, to disable flexible queuing on an MPC in slot 2:

   ```
   [edit]
   user@router# delete chassis fpc 2 flexible-queuing-mode
   ```

Copyright © 2019, Juniper Networks, Inc.

2. Review your configuration and issue the **commit** command.

```
[edit]
user@router# commit
commit complete

[edit]
user@router#
```

Related Documentation

- flexible-queuing-mode on page 1126

Multifield Classifier for Ingress Queuing on MX Series Routers with MPC

Beginning with Junos OS Release 16.1, the multifield classifier for ingress queuing is an implementation point for firewall filters configured with specific traffic shaping actions. These filters allow you to set the forwarding class and packet loss priority for packets, or drop the packets prior to ingress queue selection. The filters are applied as ingress queue filters. Class-of-service (CoS) commands can then be used to select ingress queue, set rate limiting and so forth.

Firewall filters configured at the protocol family level are able to distinguish specific types of traffic from other types by matching on multiple fields within the packet header. The number and types of matches available depend on which protocol family is used in the filter. Before the introduction of the ingress queuing filter, these firewall filters could only be applied to traffic after the ingress queue had been selected based solely on the behavior aggregate (BA). With the introduction of the ingress queuing filter, firewall filters can be used to set forwarding classification and packet loss priority based on multiple fields within the packet header prior to forwarding queue selection. CoS functions provide traffic classification options and the ability to assign that classified traffic to specific forwarding queues.

NOTE: Ingress queuing filters are only available when the traffic manager mode is set to ingress-and-egress at the **[edit chassis fpc** *fpc-id* **pic** *pic-id* **traffic-manager mode]** hierarchy level.

The **ingress-queuing-filter** configuration statement is used at the **[edit interfaces** *interface-name* **unit** *unit-number* **family** *family-name***]** hierarchy level to designate a previously configured firewall filter to be used as an ingress queuing filter. The following list shows which protocol families are compatible with the **ingress-queuing-filter** statement:

- **bridge**
- **ccc**
- **inet**
- **inet6**

Copyright © 2019, Juniper Networks, Inc.

- **mpls**
- **vpls**

The named firewall filter is a normal firewall filter that must be configured with at least one of the following actions: **accept, discard, forwarding-class,** and **loss-priority**.

Release History Table

Release	Description
16.1	Beginning with Junos OS Release 16.1, the multifield classifier for ingress queuing is an implementation point for firewall filters configured with specific traffic shaping actions.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- **ingress-queuing-filter on page 1169**
- Example: Configuring a Filter for Use as an Ingress Queuing Filter on page 945

Example: Configuring a Filter for Use as an Ingress Queuing Filter

This example shows how to configure a firewall filter for use as an ingress queuing filter. The ingress queuing filter assists in traffic shaping operations by enabling you to set the forwarding class and packet loss priority, or drop the packet before ingress queue selection. The firewall filter must be configured within one of the following protocol families: **bridge**, **cc**, **inet**, **inet6**, **mpls**, or **vpls** and have one or more of the following actions: **accept**, **discard**, **forwarding-class**, and **loss-priority**.

NOTE: Although the ingress queuing filter can be used with EX9200 switches and T-Series routers as well as MX-Series routers, it is used only on those MX Series routers that have MPCs. An error is generated at commit if the ingress queuing filter is applied to an interface on any other type of port concentrator.

- Requirements on page 946
- Overview on page 946
- Configuration on page 946

Copyright © 2019, Juniper Networks, Inc.

Requirements

This example uses the following hardware and software components:

- An MX Series router with MPC

In order for ingress queuing filters to function, **ingress-and-egress** must be configured as the **traffic-manager** mode at the **[edit chassis fpc *slot* pic *slot* traffic-manager mode]** hierarchy level.

Overview

In this example, you create a firewall filter named **iqfilter1** in the **inet** protocol family that sets the loss priority and forwarding class of packets coming from the 192.168.2.0/24 network. You then apply the **iqfilter1** filter to the ge-0/0/0.0 logical interface as an ingress queuing filter.

To configure a firewall filter and apply it for use as an ingress queuing filter involves:

- Creating a firewall filter named **iqfilter1** in the **inet** protocol family with the following two actions: **forwarding-class** and **loss-priority**.
- Applying the firewall filter to the ge-0/0/0.0 interface as an ingress queuing filter.

Configuration

- Configuring the Firewall Filter and Applying It to an Interface as an Input Queuing Filter on page 947
- Results on page 947

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.

```
set firewall family inet filter iqfilter1 term t1 from address 192.168.2.0/24
set firewall family inet filter iqfilter1 term t1 then loss-priority low
set firewall family inet filter iqfilter1 term t1 then forwarding-class expedited-forwarding
set interfaces ge-0/0/0 unit 0 family inet ingress-queuing-filter iqfilter1
```


Copyright © 2019, Juniper Networks, Inc.

Configuring the Firewall Filter and Applying It to an Interface as an Input Queuing Filter

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the firewall filter, **iqfilter1**, and apply it to logical interface ge-0/0/0 unit 0:

1. Create a firewall filter named **iqfilter1**.

```
[edit firewall family inet]
user@router# set filter iqfilter1 term t1 from address 192.168.2.0/24
user@router# set filter iqfilter1 term t1 then loss-priority low
user@router# set filter iqfilter1 term t1 then forwarding-class expedited-forwarding
```

2. Apply the firewall filter to the logical interface.

```
[edit]
user@router# set interfaces ge-0/0/0 unit 0 family inet ingress-queuing-filter
  iqfilter1
```

Results

From configuration mode, confirm your configuration by entering the **show firewall** and the **show interfaces ge-0/0/0.0** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@router# show firewall
family inet {
   filter iqfilter1 {
      term t1 {
         from {
            address {
               192.168.0.0/24;
            }
         }
         then {
            loss-priority low;
            forwarding-class expedited-forwarding;
         }
      }
   }
}
user@router# show interfaces ge-0/0/0.0
family inet {
   ingress-queuing-filter iqfilter1;
}
```

Copyright © 2019, Juniper Networks, Inc.

If you are done configuring the device, enter **commit** from configuration mode.

```
user@router# commit
```

Related Documentation

- Multifield Classifier for Ingress Queuing on MX Series Routers with MPC on page 944
- ingress-queuing-filter on page 1169

Ingress Queuing Filter with Policing Functionality

Starting with Junos OS Release 18.1R1, on MPCs that support ingress queuing, you can implement policer actions, along with other filter actions, on traffic before the traffic is assigned to ingress queues. Ingress queuing policing filters allow you to rate limit traffic as well as count and set the forwarding class and packet loss priority for packets prior to ingress queue selection. Class-of-service (CoS) commands can then be used to select ingress queuing parameters.

- Understanding the Ingress Queuing Policing Filter on page 948
- Example: Configuring a Filter for Use as an Ingress Queuing Policing Filter on page 949

Understanding the Ingress Queuing Policing Filter

The ingress queuing policing filter (**iq-policing-filter**) function similarly and at the same point as the ingress policing filter (ingress-queuing-filter), which was introduced in Junos OS Release 16.1, but provides the added benefit of accepting almost all filter actions, including policing and counting actions. The ingress queuing policing filter is also more efficient, requiring fewer system resources.

NOTE: Ingress queuing filters are only available when the traffic manager mode is set to **ingress-and-egress** at the **[edit chassis fpc *fpc-id* pic *pic-id* traffic-manager mode]** hierarchy level.

The **iq-policing-filter** configuration statement is used at the **[edit interfaces *interface-name* unit *unit-number* family *family-name*]** hierarchy level to designate a previously configured firewall filter to be used as an ingress queuing policing filter. The following list shows which protocol families are compatible with the **iq-policing-filter** statement:

- **bridge**
- **inet**
- **vpls**

See Also

- iq-policing-filter on page 1185

Copyright © 2019, Juniper Networks, Inc.

Example: Configuring a Filter for Use as an Ingress Queuing Policing Filter

This example shows how to configure a firewall filter for use as an ingress queuing policing filter. The ingress queuing filter assists in ingress traffic policing operations by allowing you to rate limit traffic prior to ingress queue selection. The firewall filter must be configured within one of the following protocol families: **bridge**, **inet**, or **vpls**.

The ingress queuing policing filter can only be used on MX Series routers with MPCs that support ingress queuing. An error is generated at commit if the ingress queuing filter is applied to an interface on any other type of port concentrator.

- Requirements on page 949
- Overview on page 949
- Configuration on page 949

Requirements

This example uses the following hardware and software components:

- An MX Series router with an MPC that supports ingress queuing

In order for ingress queuing filters to function, **ingress-and-egress** must be configured as the **traffic-manager** mode at the **[edit chassis fpc *slot* pic *slot* traffic-manager mode]** hierarchy level.

Overview

In this example, you create a firewall filter named **vpls_iqp_filter** in the **vpls** protocol family that counts and polices voice and best effort traffic. You then apply the **vpls_iqp_filter** filter to the xe-0/0/0.0 logical interface as an ingress queuing policing filter.

To configure a firewall filter and apply it for use as an ingress queuing filter involves:

- Creating a firewall filter named **vpls_iqp_filter** in the **vpls** protocol family with the following actions: **count**, **forwarding- class** and **policer**.
- Applying the firewall filter to the xe-0/0/0.0 interface as an ingress queuing policing filter.

Configuration

- Configuring the Firewall Filter and Applying It to an Interface as an Input Queuing Policing Filter on page 950
- Results on page 951

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.

```
set firewall family vpls filter vpls_iqp_filter interface-specific
set firewall family vpls filter vpls_iqp_filter term VoiceSum from learn-vlan-1p-priority 5
```

Copyright © 2019, Juniper Networks, Inc.

```
set firewall family vpls filter vpls_iqp_filter term VoiceSum then count VoiceSum
set firewall family vpls filter vpls_iqp_filter term VoiceSum then forwarding-class Voice
set firewall family vpls filter vpls_iqp_filter term VoiceSum then next term
set firewall family vpls filter vpls_iqp_filter term Voice from learn-vlan-1p-priority 5
set firewall family vpls filter vpls_iqp_filter term Voice then policer Voice-IN
set firewall family vpls filter vpls_iqp_filter term Voice then count Voice
set firewall family vpls filter vpls_iqp_filter term Voice then accept
set firewall family vpls filter vpls_iqp_filter term BestEffortSum then count BestEffortSum
set firewall family vpls filter vpls_iqp_filter term BestEffortSum then next term
set firewall family vpls filter vpls_iqp_filter term BestEffort then policer BestEffort-IN
set firewall family vpls filter vpls_iqp_filter term BestEffort then count BestEffort
set firewall family vpls filter vpls_iqp_filter term BestEffort then accept
set firewall family vpls filter vpls_iqp_filter policer pol-vpls if-exceeding bandwidth-limit
  400m
set firewall family vpls filter vpls_iqp_filter policer pol-vpls if-exceeding burst-size-limit
  40m
set firewall family vpls filter vpls_iqp_filter policer pol-vpls then discard
set firewall family vpls filter vpls_iqp_filter policer Voice-IN if-exceeding bandwidth-limit
  100m
set firewall family vpls filter vpls_iqp_filter policer Voice-IN if-exceeding burst-size-limit
  10m
set firewall family vpls filter vpls_iqp_filter policer Voice-IN then loss-priority high
set firewall family vpls filter vpls_iqp_filter policer BestEffort-IN if-exceeding
  bandwidth-limit 350m
set firewall family vpls filter vpls_iqp_filter policer BestEffort-IN if-exceeding
  burst-size-limit 30m
set firewall family vpls filter vpls_iqp_filter policer BestEffort-IN then loss-priority high
set interfaces xe-0/0/0 unit 0 family vpls iq-policing-filter vpls_iqp_filter
```

Configuring the Firewall Filter and Applying It to an Interface as an Input Queuing Policing Filter

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the firewall filter, **vpls_iqp_filter**, and apply it to logical interface xe-0/0/0 unit 0:

1. Create a firewall filter named **vpls_iqp_filter**.

```
[edit firewall family vpls filter vpls_iqp_filter]
user@router# set interface-specific
user@router# set term VoiceSum from learn-vlan-1p-priority 5
user@router# set term VoiceSum then count VoiceSum
user@router# set term VoiceSum then forwarding-class Voice
user@router# set term VoiceSum then next term
user@router# set term Voice from learn-vlan-1p-priority 5
user@router# set term Voice then policer Voice-IN
user@router# set term Voice then count Voice
user@router# set term Voice then accept
user@router# set term BestEffortSum then count BestEffortSum
user@router# set term BestEffortSum then next term
```


Copyright © 2019, Juniper Networks, Inc.

```
user@router# set term BestEffort then policer BestEffort-IN
user@router# set term BestEffort then count BestEffort
user@router# set term BestEffort then accept
user@router# set policer pol-vpls if-exceeding bandwidth-limit 400m
user@router# set policer pol-vpls if-exceeding burst-size-limit 40m
user@router# set policer pol-vpls then discard
user@router# set policer Voice-IN if-exceeding bandwidth-limit 100m
user@router# set policer Voice-IN if-exceeding burst-size-limit 10m
user@router# set policer Voice-IN then loss-priority high
user@router# set policer BestEffort-IN if-exceeding bandwidth-limit 350m
user@router# set policer BestEffort-IN if-exceeding burst-size-limit 30m
user@router# set policer BestEffort-IN then loss-priority high
```

2. Apply the firewall filter to the logical interface.

```
[edit interfaces xe-0/0/0]
user@router# set unit 0 family vpls iq-policing-filter vpls_iqp_filter
```

Results

From configuration mode, confirm your configuration by entering the **show firewall** and the **show interfaces xe-0/0/0.0** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@router# show firewall family vpls filter vpls_iqp_filter
```

```
interface-specific;
term VoiceSum {
    from {
        learn-vlan-1p-priority 5;
    }
    then {
        count VoiceSum;
        forwarding-class Voice;
        next term;
    }
}
term Voice {
    from {
        learn-vlan-1p-priority 5;
    }
    then {
        policer Voice-IN;
        count Voice;
        accept;
    }
}
term BestEffortSum {
    then {
        count BestEffortSum;
        next term;
    }
}
```

Copyright © 2019, Juniper Networks, Inc.

```
term BestEffort {
    then {
        policer BestEffort-IN;
        count BestEffort;
        accept;
    }
}
policer pol_vpls {
    if-exceeding {
        bandwidth-limit 400m;
        burst-size-limit 40m;
    }
    then discard;
}
policer Voice-IN {
    if-exceeding {
        bandwidth-limit 100m;
        burst-size-limit 10m;
    }
    then loss-priority high;
}
policer BestEffort-IN {
    if-exceeding {
        bandwidth-limit 350m;
        burst-size-limit 30m;
    }
    then loss-priority high;
}
```

```
user@router# show interfaces xe-0/0/0 unit 0
family vpls {
  iq-policing-filter vpls_iqp_filter;
}
```

If you are done configuring the device, enter **commit** from configuration mode.

```
user@router# commit
```

See Also

- iq-policing-filter on page 1185

Release History Table

Release	Description
18.1	Starting with Junos OS Release 18.1R1, on MPCs that support ingress queuing, you can implement policer actions, along with other filter actions, on traffic before the traffic is assigned to ingress queues.

Related Documentation

- iq-policing-filter on page 1185

Copyright © 2019, Juniper Networks, Inc.

Ingress Rate Limiting on MX Series Routers with MPCs

Beginning with Junos OS Release 16.2R1, on MPCs that support ingress queueing, you can perform rate limiting on incoming packets based on the forwarding class and packet loss priority (PLP) defined for each packet at ingress. You can define the ingress forwarding class either through behavior aggregate (BA) classification or through multifield (MF) ingress-queuing-filter classification.

A packed entering an interface that has ingress queuing and ingress rate-limiting enabled has the following path through the device:

```
Packet in -> BA classification -> MF classification -> Ingress rate-limiting ->
Ingress queuing -> Loopback through the interface -> BA classification -> MF
classification -> Firewall filters -> Routing -> Egress pipeline
```

As the packet enters the interface, BA and MF classification are used to determine the forwarding class and PLP for the packet. Ingress rate-limiting is applied to the packet based on the forwarding class and PLP just determined. If no BA classifier is defined for the packet, the default BA classifier is used. Use of MF classification is optional and overrides any BA classification, including default BA classification.

Ingress rate limiting can be applied to physical and logical interfaces as well as interface sets.

To configure ingress rate-limiting:

1. Configure the **traffic-manager** statement with **ingress-and-egress** mode:

   ```
   [edit chassis fpc slot-number pic pic-number]
   user@router# set traffic-manager mode ingress-and-egress;
   ```

2. Configure a rate-limited scheduler. For example:

   ```
   [edit class-of-service]
   user@router# set schedulers sched_RL transmit-rate percent 70;
   user@router# set schedulers sched_RL transmit-rate rate-limit;
   ```

3. Apply the scheduler to a scheduler map. For example:

   ```
   [edit class-of-service]
   user@router# set scheduler-maps SMap_RL forwarding-class expedited-forwarding
     scheduler sched_RL;
   ```

4. Apply the scheduler map as an **input-scheduler-map** to a physical or logical interface or interface set. For example:

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

Copyright © 2019, Juniper Networks, Inc.

```
user@router# set interfaces interface-set my-set input-scheduler-map SMap_RL;
user@router# set interfaces ge-8/0/2 input-scheduler-map SMap_RL;
user@router# set interfaces ge-8/0/3 unit 0 input-scheduler-map SMap_RL;
```

NOTE: Alternatively, apply the scheduler map to a traffic control profile, then apply the traffic control profile as an input-traffic-control-profile to a physical or logical interface or interface set.

Related Documentation

- Multifield Classifier for Ingress Queuing on MX Series Routers with MPC on page 944
- Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs on page 920

Rate Shaping on MIC and MPC Interfaces

This topic covers the following information:

- Granularity of Rate Shaping on MIC and MPC Interfaces on page 954
- Accounting for Layer 1 and Layer 2 Overhead in Egress Rate-Shaping Statistics on page 955

Granularity of Rate Shaping on MIC and MPC Interfaces

Interfaces hosted on MIC and MPC line cards have a certain granularity in the application of configured shaping rates. In other words, the observed hardware value might not exactly match the user-configured value. Nevertheless, the derived values are as close to the configured values as allowed.

Table 142 on page 954 lists the shaping granularity for each MPC port type. The derived shaping rate granularity ranges from 250 Kbps for coarse-grained queuing on the basic hardware up to 1.5 Kbps for fine-grained queuing on the enhanced queuing hardware.

Table 142: Shaping Rate Granularity for MPC Ports

MIC or MPC Port in an MX Series Router		Shaping Rate Granularity	
Line Card Type	Port Speed	Port Level, Queue Level	Logical Interface Level, Interface Set Level
Non-Queuing MPC	1 Gbps / 10 Gbps	250 Kbps	n/a
Queuing MIC or MPC	1 Gbps	2.4 Kbps	9.6 Kbps
	10 Gbps	9.6 Kbps	38.4 Kbps
Enhanced Queuing MPCs	1 Gbps	1.5 Kbps	6 Kbps
	10 Gbps	6 Kbps	24 Kbps

Copyright © 2019, Juniper Networks, Inc.

NOTE: The shaping rate granularity for MX Series routers with the MPC3E and MPC4E is approximately 293-300 Kbps. For routers with other MPCs (Trio-based FPCs), the shaping rate granularity is 250 Kbps. The predefined shaping rates for these MPCs are the next multiple of these shaping rate granularity values. The expected deviation from the predefined shaping rates is 5 to 10 percent.

Accounting for Layer 1 and Layer 2 Overhead in Egress Rate-Shaping Statistics

In calculating egress rate-shaping statistics for shaped-session packets on the egress side of MIC and MPC interfaces, the system adds 20 bytes per packet by default.

To configure an explicit overhead value to use for calculating egress rate-shaping statistics, include the egress-shaping-overhead statement at the **[edit chassis fpc *slot-number* pic *pic-number* traffic-manager]** hierarchy level. You can specify an offset value from –63 bytes through 192 bytes per egress packet.

Related Documentation

- CoS Features and Limitations on MIC and MPC Interfaces on page 930
- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972
- Scheduler Delay Buffering on MIC and MPC Interfaces on page 986
- Drop Profiles on MIC and MPC Interfaces on page 989

Per-Priority Shaping on MIC and MPC Interfaces Overview

Per-priority shaping enables you to configure a separate shaping rate for each of the five priority levels supported by MIC and MPC interfaces. The main use of per-priority shaping rates is to ensure that higher priority services such as voice and video do not starve lower priority services such as data.

There are five scheduler priorities:

- Guaranteed high (GH)
- Guaranteed medium (GM)
- Guaranteed low (GL)
- Excess high (EH)
- Excess low (EL)

The five scheduler priorities support a shaping rate for each priority:

- Shaping rate priority high (GH)
- Shaping rate priority medium (GM)
- Shaping rate priority low (GL)

Copyright © 2019, Juniper Networks, Inc.

- Shaping rate excess high (EH)
- Shaping rate excess low (EL)

On MPC7E (MPC7E-MRATE and MPC7E-10G), MPC8E (MX2K-MPC8E), MPC9E(MX2K-MPC9E), and PTX series, when you enable the enhanced priority mode feature, additional scheduler priorities and shaping rates are supported. For more information on the enhanced priority mode feature, see enhanced-priority-mode.

The additional scheduler priorities supported when you enable the enhanced priority mode feature:

- Shaping rate priority strict high (GHL)
- Shaping rate priority medium low (GML)
- Shaping rate excess medium high (EMH)
- Shaping rate excess medium low (EML)

When you enable the enhanced priority mode feature, the queue priorities are mapped to the priorities of the MPCs :

Table 143: Shaping Priority and Default Excess Shaping Priority

Configured Priority	Priority Supported on the MPC	Default Excess Priority
Strict-High	GH	EH
High	GHL	EH
Medium-High	GM	EL
Medium-Low	GML	EL
Low	GL	EM

If each service is represented by a forwarding class queued at a separate priority, then assigning a per-priority shaping rate to higher priority services accomplishes the goal of preventing the starvation of lower priority services.

To configure per-priority shaping rates, include the **shaping-rate-excess-high** *rate* <**burst-size** *burst*>, **shaping-rate-excess-low** *rate* <**burst-size** *burst*>, **shaping-rate-priority-high** *rate* <**burst-size** *burst*>, **shaping-rate-priority-low** *rate* <**burst-size** *burst*>, or **shaping-rate-priority-medium** *rate* <**burst-size** *burst*> at the **[edit class-of-service traffic-control-profiles** *tcp-name*] hierarchy level and apply the traffic control profile at the **[edit interfaces]** hierarchy level. You can specify the rate in absolute values, or by using **k** (kilo-), **m** (mega-) or **g** (giga-) units.

You can include one or more of the per-priority shaping statements in a traffic control profile:

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


Copyright © 2019, Juniper Networks, Inc.

```
traffic-control-profiles {
  tcp-ge-port {
    shaping-rate-excess-high rate <burst-size bytes>;
    shaping-rate-excess-low rate <burst-size bytes>;
    shaping-rate-priority-high rate <burst-size bytes>;
    shaping-rate-priority-low rate <burst-size bytes>;
    shaping-rate-priority-medium rate <burst-size bytes>;
  }
}
```

NOTE: To use per-priority shaping on a physical interface on the MX104 router, you must enable hierarchical scheduling on the interface with the **set hierarchical-scheduler** statement at the [**edit interface** *interface-name*] hierarchy level.

BEST PRACTICE: When planning your implementation, consider the following behavior. You can configure independent burst-size values for each rate, but the system uses the maximum burst-size value configured in each rate family. For example, the system uses the highest configured value for the guaranteed rates (GH and GM) or the highest value of the excess rates (EH and EM).

There are several important points about per-priority shaping rates:

- Per-priority shaping rates are only supported on MIC and MPC interfaces (with the exception of the 10-Gigabit Ethernet MPC with SFP+).
- Per-priority shaping is only available for level 1 and level 2 scheduler nodes. (For more information on hierarchical schedulers, see "Configuring Hierarchical Schedulers for CoS" on page 342.)
- Per-priority shaping rates are supported when level 1 or level 2 scheduler nodes have static or dynamic interfaces above them.
- Per-priority shaping rates are supported on aggregated Ethernet (AE) interfaces.
- Per-priority shaping rates are only supported in traffic control profiles.

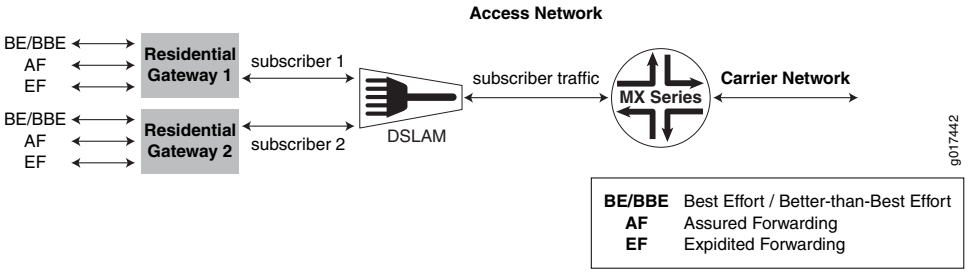
Per-priority shaping rates can be helpful when the MX Series 5G Universal Routing Platform is in a position between subscriber traffic on an access network and the carrier network, playing the role of a broadband services router. In that case, the MX Series router provides quality-of-service parameters on the subscriber access network so that each subscriber receives a minimum bandwidth (determined by the guaranteed rate) and a maximum bandwidth (determined by the shaping rate). This allows the devices closer to the carrier network to operate more efficiently and more simply and reduces operational network expenses because it allows more centralized network management.

One architecture for using per-priority shaping on the MX Series router is shown in Figure 65 on page 958. In the figure, subscribers use residential gateways with various traffic classes to support voice, video, and data services. The MX Series router sends this

Copyright © 2019, Juniper Networks, Inc.

traffic from the carrier network to the digital subscriber line access multiplexer (DSLAM) and from the DSLAM on to the residential gateway devices.

Figure 65: Architecture for MIC and MPC Interface Per-Priority Shaping



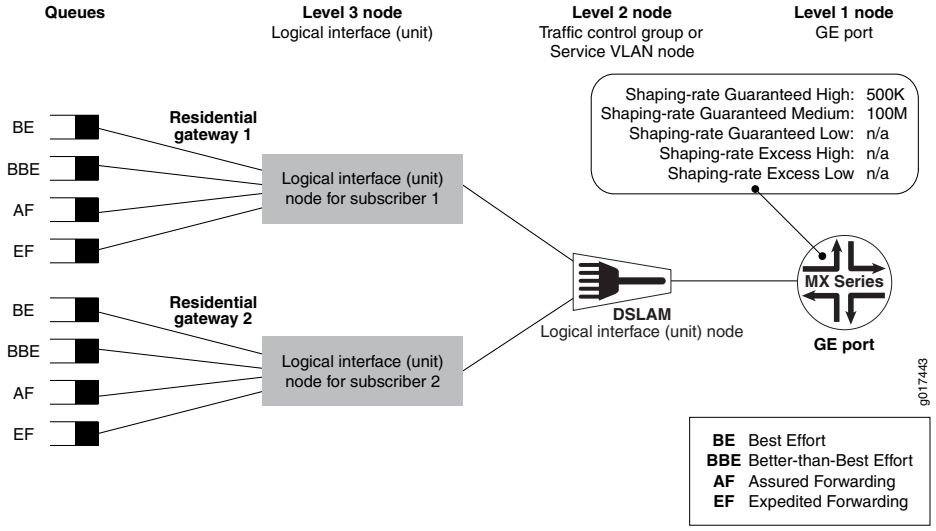
One way that the MX Series router can provide service classes for this physical network topology is shown in Figure 66 on page 959. In the figure, services such as voice and video are placed in separate forwarding classes and the services at different priority levels. For example:

- All expedited-forwarding queues are voice services at a priority level of guaranteed high.
- All assured-forwarding queues are video services at a priority level of guaranteed medium.
- All better-than-best-effort queues are services at a priority level of excess high.
- All best-effort queues are services at a priority level of excess low.

NOTE: This list covers only one possible configuration. Others are possible and reasonable, depending on the service provider's goals. For example, best-effort and better-than-best-effort traffic can have the same priority level, with the better-than-best-effort forwarding class having a higher scheduler weight than the best-effort forwarding class. For more information on forwarding classes, see "Configuring a Custom Forwarding Class for Each Queue" on page 212.

Copyright © 2019, Juniper Networks, Inc.

Figure 66: Scheduling Hierarchy for Per-Priority Shaping



Aggregated voice traffic in this topology is shaped by applying a high-priority shaper to the port. Aggregated video traffic is shaped in the same way by applying a medium-priority shaper to the port. As long as the sum of the high- and medium-priority shapers is less than the port speed, some bandwidth is reserved for best-effort and better-than-best-effort traffic. So assured-forwarding and expedited-forwarding voice and video cannot starve best-effort and better-than-best-effort data services. One possible set of values for high-priority (guaranteed high) and medium-priority (guaranteed medium) traffic is shown in Figure 66 on page 959.

BEST PRACTICE: We recommend that you do not shape delay-sensitive traffic such as voice traffic because it adds delay (latency). Service providers often use connection admission control (CAC) techniques to limit aggregated voice traffic. However, establishing a shaping rate for other traffic guards against CAC failures and can be useful in pacing extreme traffic bursts.

Per-priority shaping statements:

```
[edit class-of-service]
traffic-control-profile {
   tcp-for-ge-port {
      shaping-rate-priority-high 500k;
      shaping-rate-priority-medium 100m;
   }
}
```

Apply (attach) the traffic control profile to the physical interface (port) at the **[edit class-of-services interfaces]** hierarchy level:

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

Copyright © 2019, Juniper Networks, Inc.

```
    ge-1/0/0 {
      output-traffic-control-profile tcp-for-ge-port;
    }
}
```

Traffic control profiles with per-priority shaping rates can only be attached to interfaces that support per-priority shaping.

You can apply per-priority shaping to levels other than the level 1 physical interface (port) of the scheduler hierarchy. Per-priority shaping can also be applied at level 2, the interface set level, which would typically represent the digital subscriber link access multiplexer (DSLAM). At this level you could use per-priority shaping to limit to total amount of video traffic reaching a DSLAM, for example.

You apply (attach) the traffic control profile to an interface set at the **[edit class-of-services interfaces]** hierarchy level:

```
[edit class-of-service]
interfaces {
    interface-set svlan-1 {
      output-traffic-control-profile tcp-for-ge-port;
    }
}
```

NOTE: Although you can configure both input and output traffic control profiles, only output traffic control profiles are supported for per-priority shaping.

You can configure per-priority shaping for the traffic remaining with the **output-traffic-control-profile-remaining** statement on a physical port (a level 2 node) but not for an interface set (a level 3 node).

Release History Table

Release	Description
16.1R1	On MPC7E (MPC7E-MRATE and MPC7E-10G), MPC8E (MX2K-MPC8E), MPC9E(MX2K-MPC9E), and PTX series, when you enable the enhanced priority mode feature, additional scheduler priorities and shaping rates are supported.

Related Documentation

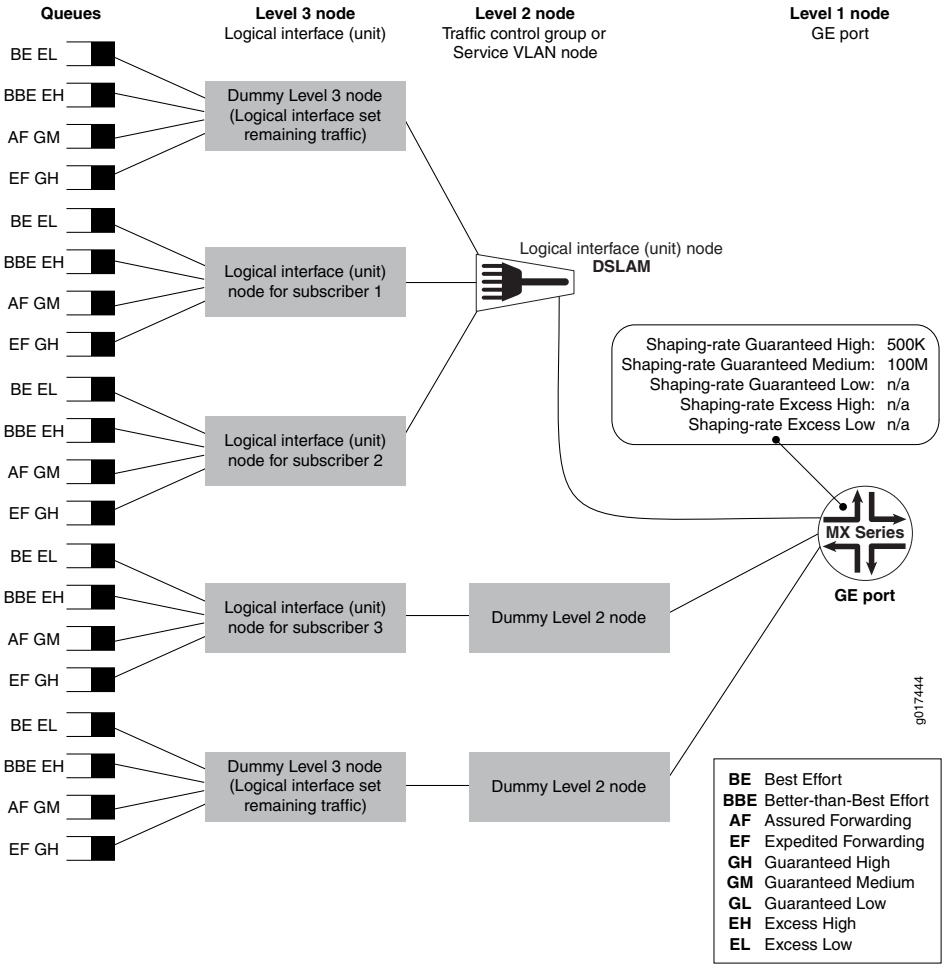
- Excess Bandwidth Distribution on MIC and MPC Interfaces Overview on page 983
- enhanced-priority-mode on page 1103

Example: Configuring Per-Priority Shaping on MIC and MPC Interfaces

In practice, per-priority shaping is used with other traffic control profiles to control traffic as a whole. Consider the traffic control profile applied to the physical interface (port), as shown in Figure 67 on page 961.

Copyright © 2019, Juniper Networks, Inc.

Figure 67: Example of MIC and MPC Interface Scheduling Hierarchy



This example is more complex than those used before. In addition to a pair of subscribers in an interface set (DSLAM), the figure now adds the following:

- A dummy level 3 scheduler node (**interface-set-remaining-traffic**) that provides scheduling for interface set members that do not have explicit class-of-service parameters configured.
- A subscriber (Subscriber 3) that is not a member of an interface set. A dummy level 2 node connects Subscriber 3's level 3 node to level 1, making it appear to be at level 2.

Copyright © 2019, Juniper Networks, Inc.

- A dummy level 3 scheduler node (**port-remaining-traffic**) in order to provide queues for traffic that does not have explicit class-of-service parameters configured.
- A dummy level 2 scheduler node to connect level 1 and level 3 scheduler nodes. This dummy level 2 scheduler node is internal only.

This example uses a gigabit Ethernet interface with five logical interface units, each one representing one of the level 3 nodes in Figure 67 on page 961.

From the top of the figure to the bottom, the level 3 nodes are:

- Unit 3 is scheduled as a "dummy" level 3 node because unit 3 is a member of an interface set (**ifset-1**) but there is no explicit CoS configuration.
- Unit 1 is scheduled as a logical interface node for subscriber 1 because unit 1 is a member of an interface set (**ifset-1**) and has an explicit CoS configuration under the **[edit class-of-service interfaces]** hierarchy.
- Unit 2 is scheduled as a logical interface node for subscriber 2 because unit 2 is a member of an interface set (**ifset-1**) and has an explicit CoS configuration under the **[edit class-of-service interfaces]** hierarchy.
- Unit 4 is scheduled as a logical interface node for subscriber 3 because unit 4 is not a member of an interface set but has an explicit CoS configuration under the **[edit class-of-service interfaces]** hierarchy level.
- Unit 5 is scheduled by another "dummy" level 3 node, this one for remaining traffic at the port level, because unit 5 is not a member of an interface set and has no explicit CoS configuration.

In this example, per-priority shaping is applied at the physical port level. The example uses three priorities, but other parameters are possible. The example does not use shaping rates, transmit rates, excess priorities, or other options for reasons of simplicity. The example uses five forwarding classes and leaves out a network control forwarding class that would typically be included in real configurations.

The example configuration is presented in several parts:

- Interfaces configuration
- Class-of-service forwarding classes and traffic control profiles configuration
- Class-of-service interfaces configuration
- Class-of-service schedulers and scheduler map configuration

Interfaces configuration:

```
[edit]
interfaces {
   # A three member interface-set.
   interface-set ifset-1 {
      interface ge-1/1/0 {
         unit 1;
         unit 2;
```

 Copyright © 2019, Juniper Networks, Inc.

```
        unit 3;
      }
    }
    # A ge port configured for "hierarchical-scheduling" and
    # vlans. 5 vlans are configured for the 5 level-3 scheduler
    # nodes
    #
    ge-1/1/0 {
      hierarchical-scheduler;
      vlan-tagging;
      unit 1 {
        vlan-id 1;
      }
      unit 2 {
        vlan-id 2;
      }
      unit 3 {
        vlan-id 3;
      }
      unit 4 {
        vlan-id 4;
      }
      unit 5 {
        vlan-id 5;
      }
    }
}
```

Class-of-service forwarding classes and traffic control profiles configuration:

```
[edit class-of-service]
forwarding-classes {
   queue 0 BE priority low;
   queue 1 BBE priority low;
   queue 2 AF priority low;
   queue 3 EF priority high;
}
traffic-control-profiles {
   tcp-if-portd {
      shaping-rate-priority-high 500k;
      shaping-rate-priority-medium 100m;
   }
   tcp-if-port-rem {
      scheduler-map smap-1;
   }
   tcp-ifset-rem {
      scheduler-map smap-1;
   }
   tcp-if-unit {
      scheduler-map smap-1;
      shaping-rate 10m;
   }
}
```

Copyright © 2019, Juniper Networks, Inc.

Class-of-service interfaces configuration:

```
[edit class-of-service]
interfaces {
   interface-set ifset-1 {
      output-traffic-control-profile-remaining tcp-ifset-rem;
   }
   ge-1/1/0 {
      output-traffic-control-profile tcp-if-port;
      output-traffic-control-profile-remaining tcp-if-port-rem;
      unit 1 {
         output-traffic-control-profile tcp-if-unit;
      }
      unit 2 {
         output-traffic-control-profile tcp-if-unit;
      }
      # Unit 3 present in the interface config and interface-set
      # config, but is absent in this CoS config so that we can
      # show traffic that uses the interface-set
      # remaining-traffic path.
      unit 4 {
         output-traffic-control-profile tcp-if-unit;
      }
      # Unit 5 is present in the interface config, but is absent
      # in this CoS config so that we can show traffic that
      # uses the if-port remaining-traffic path.
   }
}
```

Class-of-service schedulers and scheduler map configuration:

```
[edit class-of-service]
scheduler-maps {
   smap-1 {
      forwarding-class BE scheduler sched-be;
      forwarding-class BBE scheduler sched-bbe;
      forwarding-class AF scheduler sched-af;
   forwarding-class EF scheduler sched-ef;
}
schedulers {
   sched-be {
      priority low;
   }
   sched-bbe {
      priority low;
   }
   sched-af {
      priority medium-high;
   }
   sched-ef {
      priority high;
   }
}
```


Copyright © 2019, Juniper Networks, Inc.

You can configure both a shaping rate and a per-priority shaping rate. In this case, the legacy **shaping-rate** statement specifies the maximum rate for all traffic scheduled through the scheduler. Therefore, the per-priority shaping rates must be less than or equal to the overall shaping rate. So if there is a **shaping-rate 400m** statement configured in a traffic control profile, you cannot configure a higher value for a per-priority shaping rate (such as **shaping-rate-priority-high 500m**). However, the sum of the per-priority shaping rates can exceed the overall shaping rate: for **shaping-rate 400m** you can configure both **shaping-rate-priority-high 300m** and **shaping-rate-priority-low 200m** statements.

Generally, you cannot configure a shaping rate that is smaller than the guaranteed rate (which is why it is guaranteed). However, no such restriction is placed on per-priority shaping rates unless all shaping rates are for priority high or low or medium traffic.

This configuration is allowed (per-priority rates smaller than guaranteed rate):

```
[edit class-of-service]
traffic-control-profile {
  tcp-for-ge-port {
    guaranteed-rate 500m;
    shaping-rate-priority-high 400m;
    shaping-rate-priority-medium 300m;
    shaping-rate-excess-high 100m;
  }
}
```

However, this configuration generates an error (no excess per-priority rate, so the node can never achieve its guaranteed rate):

```
[edit class-of-service]
traffic-control-profile {
  tcp-for-ge-port {
    guaranteed-rate 301m;
    shaping-rate-priority-high 100m;
    shaping-rate-priority-medium 100m;
    shaping-rate-priority-low 100m;
  }
}
```

You verify configuration of per-priority shaping with the **show class-of-service traffic-control-profile** command. This example shows shaping rates established for the high and medium priorities for a traffic control profile named **tcp-ge-port**.

```
user@host# show class-of-service traffic-control-profile
```

```
Traffic control profile: tcp-ae, Index: 22093
   Shaping rate: 3000000000
   Scheduler map: <default>

Traffic control profile: tcp-ge-port, Index: 22093
   Shaping rate priority high: 1000000000
   Shaping rate priority medium: 9000000000
   Scheduler map: <default>
```

Copyright © 2019, Juniper Networks, Inc.

There are no restrictions on or interactions between per-priority shaping rates and the excess rate. An excess rate (a weight) is specified as a percentage or proportion of excess bandwidth.

Table 144 on page 966 shows where traffic control profiles containing per-priority shaping rates can be attached for both per-unit schedulers and hierarchical schedulers.

Table 144: Applying Traffic Control Profiles

Type of Traffic Control Profile	Per-unit Allowed?	Hierarchical Allowed?
Port level **output-traffic-control-profile** with per-priority shaping	Yes	Yes
Port level **output-traffic-control-profile-remaining** with per-priority shaping	No	Yes
Port level **output-traffic-control-profile** and **output-traffic-control-profile-remaining** with per-priority shaping	No	Yes
Port level **input-traffic-control-profile** with per-priority shaping	No	No
Port level **input-traffic-control-profile-remaining** with per-priority shaping	No	No
Interface set **output-traffic-control-profile** with per-priority shaping	No	Yes
Interface set **output-traffic-control-profile-remaining** with per-priority shaping	No	No
Interface set **input-traffic-control-profile** with per-priority shaping	No	No
Interface set **input-traffic-control-profile-remaining** with per-priority shaping	No	No
Logical interface level **output-traffic-control-profile** with per-priority shaping	No	No
Logical interface level **input-traffic-control-profile** with per-priority shaping	No	No

Related Documentation

- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955

Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates

The overhead accounting feature enables you to account for downstream traffic that has different encapsulations or downstream traffic from cell-based equipment, such as ATM switches.

You can configure the overhead accounting feature to shape downstream traffic based on frames or cell shaping mode.

You can also account for the different byte sizes per encapsulation by configuring a byte adjustment value for the shaping mode.

Copyright © 2019, Juniper Networks, Inc.

To configure the shaping mode and byte adjustment value for static CoS configurations:

1. Specify the shaping mode.

   Frame shaping mode is enabled by default.

   ```
   [edit class-of-service traffic-control-profiles profile-name]
   user@host# set overhead-accounting (frame-mode | cell-mode)
   ```

2. (Optional) Specify a byte adjustment value.

   ```
   [edit class-of-service traffic-control-profiles profile-name
   user@host# set overhead-accounting bytes byte-value]
   ```

BEST PRACTICE: We recommend that you specify a byte adjustment value that represents the difference between the customer premise equipment (CPE) protocol overhead and the B-RAS protocol overhead.

The available range is –120 through 124 bytes. The system rounds up the byte adjustment value to the nearest multiple of 4. For example, a value of 6 is rounded to 8, and a value of –10 is rounded to –8.

Related Documentation

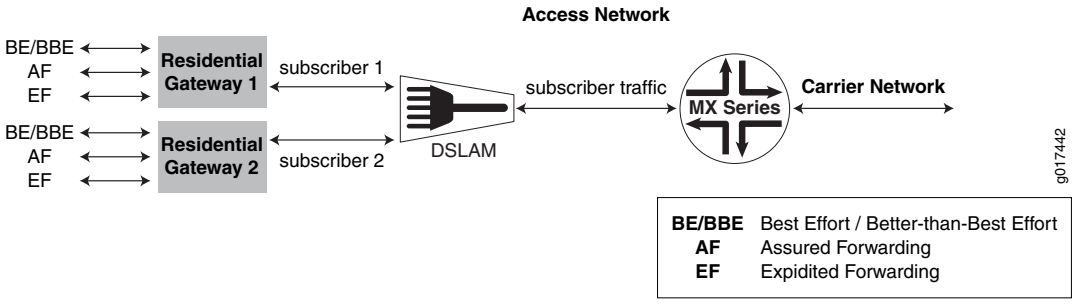
- Bandwidth Management for Downstream Traffic in Edge Networks Overview on page 984

Example: Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates

This topic describes two scenarios for which you can configure static shaping parameters to account for packet overhead in a downstream network.

Figure 68 on page 967 shows the sample network that the examples reference.

Figure 68: Sample Network Topology for Downstream Traffic



Managing Traffic with Different Encapsulations

In this example, the MX Series router shown in Figure 68 on page 967 sends stacked VLAN frames to the DSLAM, and the DSLAM sends single-tagged VLAN frames to the residential gateway.

To accurately shape traffic at the residential gateway, the MX Series router must account for the different frame sizes. The difference between the stacked VLAN (S-VLAN) frames sent by the router and the single-tagged VLAN frames received at the residential gateway is a 4-byte VLAN tag. The residential gateway receives frames that are 4 bytes less.

To account for the different frame sizes, the network administrator configures the frame shaping mode with –4 byte adjustment:

1. The network administrator configure the traffic shaping parameters and attaches them to the interface.

   Enabling the overhead accounting feature affects the resulting shaping rate, guaranteed rate, and excess rate parameters, if they are configured.

```
[edit]
class-of-service {
  traffic-control-profiles {
    tcp-example-overhead-accounting-frame-mode {
      shaping-rate 10m;
      shaping-rate-priority-high 4m;
      guaranteed-rate 2m;
      excess-rate percent 50;
      overhead-accounting frame-mode bytes -4;
      }
    }
    interfaces {
      ge-1/0/0 {
        output-traffic-control-profile tcp-example-overhead-accounting-frame-mode;
      }
    }
  }
}
```

2. The network administrator verifies the adjusted rates.

```
user@host#show class-of-service traffic-control-profile
```

```
Traffic control profile: tcp-example-overhead-accounting-frame-mode, Index:
61785
Shaping rate: 10000000
Shaping rate priority high: 4000000
Excess rate 50
Guaranteed rate: 2000000
Overhead accounting mode: Frame Mode
Overhead bytes: –4
```


Copyright © 2019, Juniper Networks, Inc.

Managing Downstream Cell-Based Traffic

In this example, the DSLAM and residential gateway shown in Figure 68 on page 967 are connected through an ATM cell-based network. The MX Series router sends Ethernet frames to the DSLAM, and the DSLAM sends ATM cells to the residential gateway.

To accurately shape traffic at the residential gateway, the MX Series router must account for the different physical network characteristics.

To account for the different frame sizes, the network administrator configures the cell shaping mode with –4 byte adjustment:

1. Configure the traffic shaping parameters and attach them to the interface.

   Enabling the overhead accounting feature affects the resulting shaping rate, guaranteed rate, and excess rate parameters, if they are configured.

```
[edit]
class-of-service {
  traffic-control-profiles {
    tcp-example-overhead-accounting-cell-mode {
      shaping-rate 10m;
      shaping-rate-priority-high 4m;
      guaranteed-rate 2m;
      excess-rate percent 50;
      overhead-accounting cell-mode;
      }
    }
    interfaces {
      ge-1/0/0 {
        output-traffic-control-profile tcp-example-overhead-accounting-cell-mode;
      }
    }
  }
}
```

2. Verify the adjusted rates.

```
user@host#show class-of-service traffic-control-profile
```

```
Traffic control profile: tcp-example-overhead-accounting-cell-mode, Index:
61785
Shaping rate: 10000000
Shaping rate priority high: 4000000
Excess rate 50
Guaranteed rate: 2000000
Overhead accounting mode: Cell Mode
Overhead bytes: 0
```

   To account for ATM segmentation, the MX Series router adjusts all of the rates by 48/53 to account for ATM AAL5 encapsulation. In addition, the router accounts for cell padding, and internally adjusts each frame by 8 bytes to account for the ATM trailer.

Copyright © 2019, Juniper Networks, Inc.

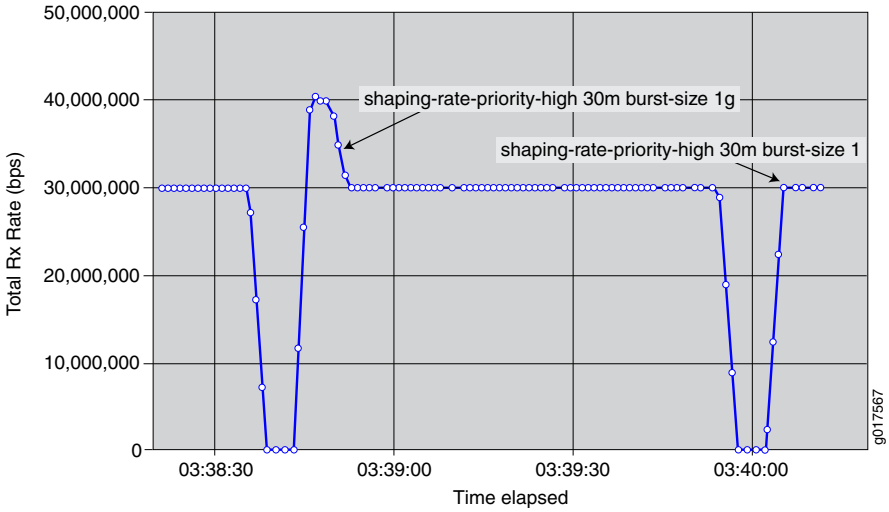
Related Documentation

- Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates on page 966

Traffic Burst Management on MIC and MPC Interfaces Overview

You can manage the impact of bursts of traffic on your network by configuring a burst-size value with the shaping rate or the guaranteed rate. The value is the maximum bytes of rate credit that can accrue for an idle queue or scheduler node. When a queue or node becomes active, the accrued rate credits enable the queue or node to catch up to the configured rate.

Figure 69: Sample Burst Shaping Rates



In Figure 69 on page 970, the network administrator configures a large burst-size value for the shaping rate, then configures a small burst-size value. The larger burst size is subject to a maximum value. The smaller burst size is subject to a minimum value that enables the system to achieve the configured rates.

In both configurations, the scheduler node can burst beyond its shaping rate for a brief interval. The burst of traffic beyond the shaping rate is more noticeable with the larger burst size than the smaller burst size.

- Guidelines for Configuring the Burst Size on page 970
- How the System Calculates the Burst Size on page 971

Guidelines for Configuring the Burst Size

Typically, the default burst-size (100 ms) for both scheduler nodes and queues on MIC and MPC interfaces is adequate for most networks. However, if you have intermediate equipment in your network that has very limited buffering and is intolerant of bursts of traffic, you might want to configure a lower value for the burst size.

Use caution when selecting a different burst size for your network. A burst size that is too high can overwhelm downstream networking equipment, causing dropped packets and

Copyright © 2019, Juniper Networks, Inc.

inefficient network operation. Similarly, a burst size that is too low can prevent the network from achieving your configured rate.

When configuring a burst size, keep the following considerations in mind:

- The system uses an algorithm to determine the actual burst size that is implemented for a node or queue. For example, to reach a shaping rate of 8 Mbps, you must allocate 1Mb of rate credits every second. A shaping rate of 8 Mbps with a burst size of 500,000 bytes of rate-credit per seconds enables the system to transmit at most 500,000 bytes, or 4 Mbps. The system cannot implement a burst size that prevents the rate from being achieved.

  For more information, see "How the System Calculates the Burst Size" on page 971.

- There are minimum and maximum burst sizes for each platform, and different nodes and queue types have different scaling factors. For example, the system ensures the burst cannot be set lower than 1 Mbps for a shaping rate of 8 Mbps. To smoothly shape traffic, rate credits are sent much faster than once per second. The interval at which rate credits are sent varies depending on the platform, the type of rate, and the scheduler level.

- When you have configured adjustments for the shaping rate (either by percentage or through an application such as ANCP or Multicast OIF), the system bases the default and minimum burst-size calculations on the adjusted shaping rate.

- When you have configured cell shaping mode to account for ATM cell tax, the system bases the default and minimum burst-size calculations on the post-tax shaping rate.

- The guaranteed rate and shaping rate share the value specified for the burst size. If the guaranteed rate has a burst size specified, that burst size is used for the shaping rate; if the shaping rate has a burst size specified, that bursts size is used for the guaranteed rate. If you have specified a burst size for both rates, the system uses the lesser of the two values.

- The burst size configured for the guaranteed rate cannot exceed the burst-size configured for the shaping rate. Starting in Junos OS Release 15.1, the CLI no longer generates a commit error when the guaranteed-rate burst size is statically configured to be more than the shaping-rate burst size. This behavior changed with the advent of enhanced subscriber management. The system logs an error when the guaranteed-burst rate is higher, whether it is configured statically, dynamically with predefined variables, or by means of a change of authorization request.

- If you have not configured a guaranteed rate, logical interfaces and interface sets receive a default guaranteed rate from the port speed. Queues receive a default guaranteed rate from the parent logical interface or interface set.

- Burst-size is not supported with **per-priority-shaping**.

How the System Calculates the Burst Size

When calculating the burst size, the system uses an exponent of a power of two. For example:

Shaping-rate in bps * 100 ms / (8 bits/byte * 1000 ms/s) = 1,875,000 bytes

Copyright © 2019, Juniper Networks, Inc.

The system then rounds this value up. For example, the system uses the following calculation to determine the burst size for a scheduler node with a shaping rate of 150 Mbps:

Max (Shaping rate, Guaranteed rate) bps * 100 ms / (8 bits/byte * 1000 ms/s) = 1,875,000 bytes

Rounded up to the next higher power of two = 2,097,150 (which is 221, or 0x200000)**

The system assigns a single burst size to each of the following rate pairs:

- Shaping rate and guaranteed rate
- Guaranteed high (GH) and guaranteed medium (GM)
- Excess high (EH) and excess low (EL)
- Guaranteed low (GL)

To calculate the burst size for each pair, the system:

- Uses the configured burst-size if only one of the pair is configured.
- Uses the lesser of the two burst sizes if both values are configured.
- Uses the next lower power of two.
- To calculate the minimum burst size, the system uses the greater of the two rates.

Release History Table

Release	Description
15.1	Starting in Junos OS Release 15.1, the CLI no longer generates a commit error when the guaranteed-rate burst size is statically configured to be more than the shaping-rate burst size.

Related Documentation

- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 986

Understanding Hierarchical Scheduling for MIC and MPC Interfaces

This topic covers the following information:

- Scheduler Node Scaling for MIC and MPC Interfaces on page 973
- Hierarchical Scheduling Priority Levels for MIC and MPC Interfaces on page 973
- Guaranteed Bandwidth and Weight of an Interface Node on MIC and MPC Interfaces on page 973
- Hierarchical Scheduling for MIC and MPC Interfaces in Oversubscribed PIR Mode on page 974

Copyright © 2019, Juniper Networks, Inc.

Scheduler Node Scaling for MIC and MPC Interfaces

In per-unit scheduling, the logical interfaces share a common level 2 node (one per port). In hierarchical-scheduling, each logical interface has its own level 2 node. Thus, scaling is limited by the number of level 2 nodes.

To better control system resources in hierarchical-scheduling mode, you can limit the number of scheduler node levels to two. In this case, all logical interfaces and interface sets with CoS scheduling policy share a single level 2 node. Consequently, the maximum number of logical interfaces with CoS scheduling policies is increased (the interface sets must be at level 3).

To configure scheduler node scaling, include the **hierarchical-scheduler** statement and set the **maximum-hierarchy-levels** option to **2** at the **[edit interfaces xe-*fpc/pic/port*]** hierarchy level.

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

NOTE: The maximum-hierarchy-levels option supports level 3 interface sets but not level 2 interface sets. If you configure level 2 interface sets with the maximum-hierarchy-levels option, you generate Packet Forwarding Engine errors.

Hierarchical Scheduling Priority Levels for MIC and MPC Interfaces

The queuing model used by MIC and MPC interfaces supports three priority levels for guaranteed scheduling priority and two lower priority levels for excess scheduling priority. You can configure a queue with one guaranteed priority and one excess priority. For example, you can configure a queue for guaranteed low (GL) as the guaranteed priority and configure excess high (EH) as the excess priority.

You can associate a guaranteed level with only one excess level. You can associate an excess level with any number of guaranteed priority levels, including none.

Interface nodes maintain their guaranteed priority level (for example, guaranteed high, GH) as long as they do not exceed their guaranteed bandwidth. If the queue bandwidth exceeds the guaranteed rate, then the priority drops to the excess priority (for example, excess high, EH). Because excess level priorities are lower than their guaranteed counterparts, the bandwidth guarantees for each of the other levels can be maintained.

Guaranteed Bandwidth and Weight of an Interface Node on MIC and MPC Interfaces

The queuing model used by MIC and MPC interfaces separates the concepts of *guaranteed bandwidth* and *weight* of an interface node, although the two terms are often used interchangeably. The guaranteed bandwidth for an interface node is the bandwidth the

Copyright © 2019, Juniper Networks, Inc.

node can use, independent of what is happening at the other nodes of the scheduling hierarchy. The weight of an interface node, on the other hand, is a value that determines how *excess bandwidth* is used. The weight of a node comes into play when other nodes at the same hierarchical scheduling level use less than the sum of their guaranteed bandwidths

For some application traffic types (such as constant bit rate voice, where there is little concern about excess bandwidth), the guaranteed bandwidth dominates the node. For other types of application traffic (such as bursty data, where a well-defined bandwidth is not always possible), the concept of weight dominates the node.

Hierarchical Scheduling for MIC and MPC Interfaces in Oversubscribed PIR Mode

In contrast to the Intelligent Queuing Enhanced (IQE) and Intelligent Queuing 2 Enhanced (IQ2E) PICs, the interfaces on MICs and MPCs set the guaranteed rate to zero in oversubscribed peak information rate (PIR) mode for the per-unit scheduler. Also, the configured rate is scaled down to fit the oversubscribed value. For example, if there are two logical interface units with a shaping rate of 1 Gbps each on a 1-Gbps port (which is, therefore, oversubscribed 2 to 1), then the guaranteed rate on each unit is scaled down to 500 Mbps (scaled down by 2).

With hierarchical schedulers in oversubscribed PIR mode, the guaranteed rate for every logical interface unit is set to zero. This means that the queue transmit rates are always oversubscribed.

Because in oversubscribed PIR mode the queue transmit rates are always oversubscribed, the following are true:

- If the queue transmit rate is set as a percentage, then the guaranteed rate of the queue is set to zero; but the excess rate (weight) of the queue is set correctly.
- If the queue transmit rate is set as an absolute value and if the queue has guaranteed high or medium priority, then traffic up to the queue's transmit rate is sent at that priority level. However, for guaranteed low traffic, that traffic is demoted to the excess low region. This means that best-effort traffic well within the queue's transmit rate gets a lower priority than out-of-profile excess high traffic. This differs from the IQE and IQ2E PICs.

Related Documentation

- Per-Unit Queuing and Hierarchical Queuing for MIC and MPC Interfaces on page 978
- CoS Features and Limitations on MIC and MPC Interfaces on page 930
- Jitter Reduction in Hierarchical CoS Queues on page 991
- Scheduling and Shaping in Hierarchical CoS Queues for Traffic Routed to GRE Tunnels on page 576
- *CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces*

Copyright © 2019, Juniper Networks, Inc.

Configuring Ingress Hierarchical CoS on MIC and MPC Interfaces

You can configure ingress CoS parameters, including hierarchical schedulers, on MIC and MPC interfaces on MX Series routers. In general, the supported configuration statements apply to per-unit schedulers or to hierarchical schedulers.

NOTE: Ingress CoS is not supported on AE interfaces on MPCs.

To configure ingress CoS for per-unit schedulers, include the following statements at the **[edit class-of-service interfaces *interface-name*]** hierarchy level:

```
[edit class-of-service interfaces interface-name]
input-excess-bandwidth-share (proportional value | equal);
input-scheduler-map map-name;
input-shaping-rate rate;
input-traffic-control-profile profile-name;
unit logical-unit-number;
  input-scheduler-map map-name;
  input-shaping-rate (percent percentage | rate);
  input-traffic-control-profile profile-name;
```

To configure ingress CoS for hierarchical schedulers, include the **interface-set *interface-set-name*** statement at the **[edit class-of-service interfaces]** hierarchy level:

```
[edit class-of-service interfaces]
interface-set interface-set-name {
  input-traffic-control-profile profile-name;
  input-traffic-control-profile-remaining profile-name;
  interface interface-name {
    input-excess-bandwidth-share (proportional value | equal);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    unit logical-unit-number;
  }
}
```

By default, ingress CoS features are disabled on MIC and MPC interfaces. To enable ingress CoS on a MIC or MPC interface, configure the **traffic-manager** statement with **ingress-and-egress** mode as shown in the following example:

```
chassis {
  fpc 7 {
    pic 0 {
      traffic-manager {
        mode ingress-and-egress;
      }
    }
  }
}
```

Configured CoS features on the ingress are independent of CoS features on the egress.

Copyright © 2019, Juniper Networks, Inc.

NOTE: Prior to Junos OS 16.1R1, for MIC-based MX80 and MX104 routers, only ten queues on one MIC can be configured for ingress CoS. Starting with Junos OS 16.1R1, MX80 and MX104 routers support up to 12 ingress queues on any combination of both MIC and built-in ports.

Starting with Junos 17.4R2 on MX80 and MX104 routers, you can have precise control over which ports have ingress CoS enabled by configuring **traffic-manager** at the port level (**[edit chassis fpc** *slot-number* **pic** *pic-number* **port** *port-number***]** hierarchy level). You cannot, however, configure **traffic-manager** at both the port level and PIC level on the same device.

The following behavior aggregate (BA) classification tables are supported on the ingress side of MIC and MPC interfaces:

- DSCP
- DSCP for IPv6
- exp (MPLS)
- IEEE 802.1p
- inet-precedence

Release History Table

Release	Description
17.4R2	Starting with Junos 17.4R2 on MX80 and MX104 routers, you can have precise control over which ports have ingress CoS enabled by configuring **traffic-manager** at the port level
16.1R1	Starting with Junos OS 16.1R1, MX80 and MX104 routers support up to 12 ingress queues on any combination of both MIC and built-in ports.

Related Documentation

- mode (Layer 2 Tunneling Protocol Shaping) on page 1211

Copyright © 2019, Juniper Networks, Inc.

Configuring a CoS Scheduling Policy on Logical Tunnel Interfaces

You can configure a CoS scheduling policy on a logical tunnel interface (LT ifl). Logical tunnel interfaces can be used to terminate a pseudowire into a virtual routing and forwarding (VRF) instance. If an lt device is used to terminate a pseudowire, CoS scheduling policies can be applied on the lt interface to manage traffic entering the pseudowire. You accomplish this by configuring the hierarchical-scheduler attribute on the physical interface.

NOTE: It is important to first commit the hierarchical-scheduler configuration under the logical tunnel physical interface (LT ifd), and subsequently add and commit the class-of-service configuration.

NOTE: The output-traffic-control statement applies only to the LT ifl that is part of an L3 VRF instance.

The following example shows two pseudowires (pw1 and pw2) over lt-1/0/10. pw1 carries data, voice, and video traffic, and pw2 carries only data and voice traffic. All pseudowire traffic is restricted to 800m bps. The shaping rate for traffic over pw1 is 400m bps and the shaping rate for traffic over pw2 is 400m bps.

```
[edit interfaces]
lt-1/0/10 {
   hierarchical-scheduler;
}
[edit class-of-service schedulers]
data_sch {
   buffer-size remainder;
   priority low;
}
voice_sch {
   transmit-rate 6k;
   priority strict-high;
}
video_sch {
   shaping-rate 1m;
   priority medium-low;
}
[edit class-of-service scheduler-maps]
pw1-smap {
   forwarding-class be scheduler data_sch;
   forwarding-class ef scheduler voice_sch;
   forwarding-class af scheduler video_sch;
}
pw2-smap {
   forwarding-class be scheduler data_sch;
   forwarding-class ef scheduler voice_sch;
}
```

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service traffic-control-profiles]
pw1-tcp {
   scheduler-map pw1-smap;
   shaping-rate 400m;
}
pw2-tcp {
   scheduler-map pw2-smap;
   shaping-rate 400m;
}
all-pw-tcp {
   shaping-rate 800m;
}
lt-ifd-remain {
   shaping-rate 10m;
}
[edit class-of-service interfaces]
lt-1/0/10 {
   output-traffic-control-profile all-pw-tcp;
   output-traffic-control-profile-remaining lt-ifd-remain;
unit 1 {
   output-traffic-control-profile pw1-tcp;
}
unit 3 {
   output-traffic-control-profile pw2-tcp;
}
```

Related Documentation

- CoS Scheduling Policy on Logical Tunnel Interfaces Overview on page 1000
- Configuring Hierarchical Schedulers for CoS on page 342
- *Configuring Logical Tunnel Interfaces*
- CoS on Ethernet Pseudowires in Universal Edge Networks Overview on page 999
- Configuring CoS on an Ethernet Pseudowire for Multiservice Edge Networks on page 1000

Per-Unit Queuing and Hierarchical Queuing for MIC and MPC Interfaces

This topic covers the following information:

- Queuing Models Supported for MIC and MPC Interfaces on page 978
- Scheduler Node Levels for MIC and MPC Interfaces on page 980

Queuing Models Supported for MIC and MPC Interfaces

Interfaces hosted on Modular Interface Card (MIC) and Modular Port Concentrator (MPC) line cards in MX Series 5G Universal Routing Platforms support the following models of class-of-service (CoS) queuing, depending on MIC or MPC type:

- Limited Scale Per-Unit Queuing MPCs on page 979
- Hierarchical Queuing MICs and MPCs on page 979

Copyright © 2019, Juniper Networks, Inc.

Limited Scale Per-Unit Queuing MPCs

Per-unit CoS queuing features on a limited scale are supported for interfaces hosted on some MPCs that do not have a dedicated queuing chip, specifically the MPC3E, MPC4E, MPC5E, and MPC6E line cards and on the fixed-configuration 16-port 10-Gigabit Ethernet MPC in MX240, MX480, MX960, MX2010, and MX2020 routers.

NOTE: **The nonqueuing MPC1 and MPC2 line cards *do not* support per-unit queuing.**

On MPCs that support per-unit queuing, the following queuing capabilities are available:

- Four or eight egress queues per unit.
- Delay buffer capacities of 100 ms by default, and up to 200 ms maximum delay.
- Rate shaping of the ports and their queues.
- Guaranteed rate enforced at the queues.

The per-unit CoS queuing features also support pre-classification of incoming packets to protect high priority packets in the event of congestion. Such features include ingress DSCP rewrite and per-VLAN classification, ingress and egress policing, and rewrites.

Hierarchical Queuing MICs and MPCs

Hierarchical CoS queuing features are supported on interfaces hosted on MICs in MPC1 Q, MPC2 Q, MPC2 EQ, MPC5EQ, MPC7E, MPC8E, and MPC9E line cards in MX204, MX240, MX480, MX960, MX2010, MX2020, and MX10003 routers and for interfaces hosted on 1-Gigabit and 10-Gigabit Ethernet MICs in MX5, MX10, MX40, MX80, or MX104 modular chassis routers. These MICs and MPCs provide a dedicated queuing chip that supports hierarchical queuing.

Hierarchical queuing MICs and MPCs support all per-unit queuing functionality plus fine-grained queuing abilities over four or five levels of hierarchical scheduling:

- Hierarchical scheduling with ports, interface sets, and logical interfaces.
- Shaping—Committed Information Rate (CIR) and a peak information rate (PIR)—at all scheduling levels, including queues.
- Three normal- priority levels and two excess- priority levels configurable at all scheduling levels, including queues.
- Per-priority shaping of traffic at Level 1 or Level 2.
- Shaping for unconfigured customer VLANs (C-VLANs) and for service VLANs (S-VLANs).

Copyright © 2019, Juniper Networks, Inc.

Scheduler Node Levels for MIC and MPC Interfaces

Interfaces hosted on MICs and MPCs support different scheduler node levels, depending on MIC or MPC type:

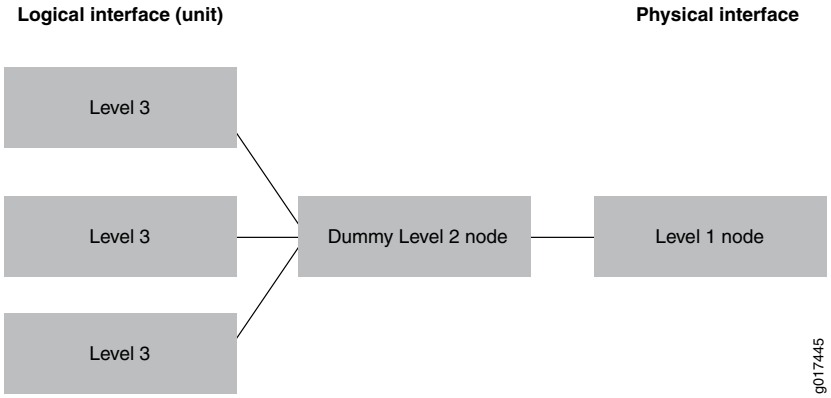
- Scheduler Node Levels for Per-Unit Queuing MPCs on page 980
- Scheduler Node Levels for Hierarchical Queuing MICs and MPCs on page 980

Scheduler Node Levels for Per-Unit Queuing MPCs

For an interface hosted on a per-unit queuing MPC, each logical interface has its own dedicated level 3 node, and all logical interfaces share a common level 2 node (one per port).

Figure 70 on page 980 illustrates scheduler node levels for an interface hosted on a per-unit queuing MPC.

Figure 70: Scheduler Node Levels for Per-Unit Queuing MPCs



For interfaces hosted on per-unit queuing MPCs, the level 2 node is always a dummy node.

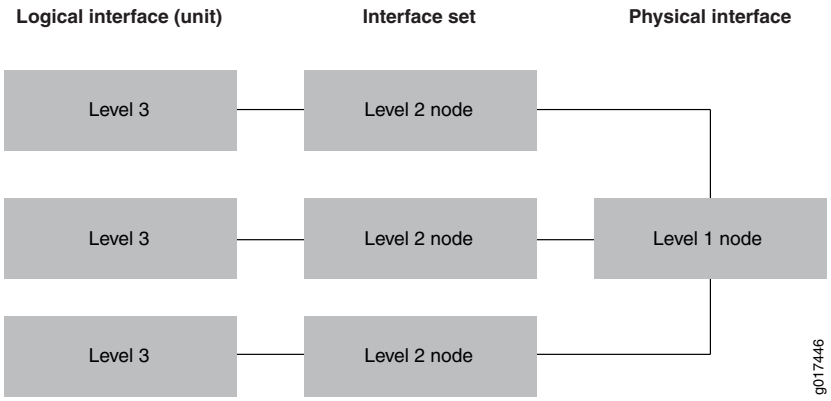
Scheduler Node Levels for Hierarchical Queuing MICs and MPCs

With the exception of the 10-Gigabit Ethernet MPC with SFP+, the queuing model used by interfaces hosted on hierarchical queuing MICs and MPCs supports up to five levels of scheduler nodes: the queue itself (level 5), session logical interface (ppp or dhcp) (level 4), customer VLAN (C-VLAN) (level 3), the interface set or service VLAN (S-VLAN) collection (level 2), and the physical interface or port (level 1).

Figure 71 on page 981 illustrates the scheduler node levels for an interface hosted on a hierarchical queuing MIC or MPC.

Copyright © 2019, Juniper Networks, Inc.

Figure 71: Scheduler Node Levels for Interfaces on Hierarchical Queuing and Scheduling MICs and MPCs



The figure depicts scheduler nodes for an interface that does not include interface sets and for which traffic control profiles are applied to the logical interfaces only.

NOTE: If an interface set has a CoS scheduling policy but none of its child logical interfaces has a CoS scheduling policy, then the interface set is considered to be a leaf node and has one level 2 and one level 3 node.

Related Documentation

- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972
- *CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces*
- *MX Series MPC Overview*
- *MPCs Supported by MX Series Routers*
- *MX Series MIC Overview*
- *MICs Supported by MX Series Routers*
- *MX5, MX10, MX40, and MX80 Modular Interface Card Description*

Managing Dedicated and Remaining Queues for Static CoS Configurations on MIC and MPC Interfaces

This topic describes how to manage dedicated and remaining queues for static subscriber interfaces configured at the **[edit class-of-service]** hierarchy.

- Configuring the Maximum Number of Queues for MIC and MPC Interfaces on page 981
- Configuring Remaining Common Queues on MIC and MPC Interfaces on page 982

Configuring the Maximum Number of Queues for MIC and MPC Interfaces

30-Gigabit Ethernet Queuing MPCs and 60-Gigabit Ethernet Queuing and Enhanced Queuing MPCs support a dedicated number of queues when configured for hierarchical scheduling and per-unit scheduling configurations.

Copyright © 2019, Juniper Networks, Inc.

To scale the number of subscriber interfaces per queue, you can modify the number of queues supported on the MIC.

To configure the number of queues:

1. Specify that you want to configure the MIC.

```
user@host# edit chassis fpc slot-number pic pic-number
```

2. Configure the number of queues.

```
[edit chassis fpc slot-number pic pic-number]
user@host# set max-queues-per-interface (8 | 4)
```

See Also

- Dedicated Queue Scaling for CoS Configurations on MIC and MPC Interfaces Overview on page 931

Configuring Remaining Common Queues on MIC and MPC Interfaces

30-Gigabit Ethernet Queuing MPCs and 60-Gigabit Ethernet Queuing and Enhanced Queuing MPCs support a dedicated set of queues when configured with hierarchical scheduling.

When the number of dedicated queues is reached on the module, there can be queues remaining. Traffic from these logical interfaces are considered unclassified and attached to a common set of queues that are shared by all subsequent logical interfaces.

You can configure traffic shaping and scheduling resources for the remaining queues by attaching a special traffic-control profile to the interface. This feature enables you to provide the same shaping and scheduling to remaining queues as the dedicated queues.

To configure the remaining queues on a MIC or MPC interface:

1. Configure CoS parameters in a traffic-control profile.

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

2. Enable hierarchical scheduling for the interface.

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

3. Attach the traffic control profiles for the dedicated and remaining queues to the port on which you enabled hierarchical scheduling.

Copyright © 2019, Juniper Networks, Inc.

To provide the same shaping and scheduling parameters to dedicated and remaining queues, reference the same traffic-control profile.

a. Attach the traffic-control profile for the dedicated queues on the interface.

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

b. Attach the traffic-control profile for the remaining queues on the interface.

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

See Also

- Dedicated Queue Scaling for CoS Configurations on MIC and MPC Interfaces Overview on page 931

Related Documentation

- Dedicated Queue Scaling for CoS Configurations on MIC and MPC Interfaces Overview on page 931
- Verifying the Number of Dedicated Queues Configured on MIC and MPC Interfaces on page 934
- Configuring Hierarchical Schedulers for CoS on page 342
- Configuring Interface Sets on page 264

Excess Bandwidth Distribution on MIC and MPC Interfaces Overview

Service providers often used tiered services to provide bandwidth for excess traffic as traffic patterns vary. By default, excess bandwidth between a configured guaranteed rate and shaping rate is shared equally among all queues on MIC and MPC interfaces, which might not be optimal for all subscribers to a service.

You can adjust this distribution by configuring the rates and priorities for the excess bandwidth.

By default, when traffic exceeds the shaping or guaranteed rates, the system demotes traffic with guaranteed high (GH) priority and guaranteed medium (GM) priority. You can disable this priority demotion for the MIC and MPC interfaces in your router.

Related Documentation

- Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 986
- *Managing Excess Bandwidth Distribution for Dynamic CoS on MIC and MPC Interfaces*
- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- Traffic Burst Management on MIC and MPC Interfaces Overview on page 970

Copyright © 2019, Juniper Networks, Inc.

Bandwidth Management for Downstream Traffic in Edge Networks Overview

In a subscriber access network, traffic with different encapsulations can be passed downstream to other customer premise equipment (CPE) through the MX Series router. Managing the bandwidth of downstream ATM traffic to Ethernet interfaces can be especially difficult because of the different Layer 2 encapsulations.

The downstream network is not necessarily the directly attached network. In typical broadband network gateway (BNG) configurations, the directly attached network is an Ethernet access network, which provides access to either another frame-based network, or a cell-based network.

The *overhead accounting* feature enables you to shape traffic based on whether the downstream network is a frame-based network, like Ethernet, or a cell-based network, like ATM. It assigns a byte adjustment value to account for different encapsulations.

This feature is available on MIC and MPC interfaces.

Effective Shaping Rate

The shaping-rate, also known as peak information rate (PIR), is the maximum rate for a scheduler node or queue.

The true rate of a subscriber at the access-loop/CPE is a function of:

- The shaping-rate in effect for the subscriber's household, in bits per second.
- Whether the subscriber is connected to a frame-based or cell-based network.
- Number of bytes in each frame that are accounted for by the shaper.

NOTE: Chassis egress-shaping-overhead is not included in the effective rate. Egress-shaping-overhead accounts for the physical interface overhead (ISO OSI Layer 1). Effective shaping-rate is a Layer 2 (ISO OSI) rate.

Shaping Modes

There are two modes used for adjusting downstream traffic:

- *Frame shaping mode* is useful for adjusting downstream traffic with different encapsulations. Shaping is based on the number of bytes in the frame, without regard to cell encapsulation or padding overhead. Frame is the default shaping mode on the router.
- *Cell shaping mode* is useful for adjusting downstream cell-based traffic. In cell shaping mode, shaping is based on the number of bytes in cells, and accounts for the cell encapsulation and padding overhead.

  When you specify cell mode, the resulting traffic stream conforms to the policing rates configured in downstream ATM switches, reducing the number of packet drops in the Ethernet network.

Copyright © 2019, Juniper Networks, Inc.

To account for ATM segmentation, the router adjusts all of the rates by 48/53 to account for 5-byte ATM AAL5 encapsulation. In addition, the router accounts for cell padding, and internally adjusts each frame by 8 bytes to account for the ATM trailer.

Byte Adjustments

When the downstream traffic has different byte sizes per encapsulation, it is useful to configure a *byte adjustment* value to adjust the number of bytes per packet to be included in or excluded from the shaping mechanism. This value represents the number of bytes that are encapsulated and decapsulated by the downstream equipment. For example, to properly account for a 4-byte header stripped by the downstream network, set the overhead-accounting bytes to -4. To properly account for a 12-byte header added by the downstream network, set the overhead-accounting bytes to 12.

We recommend that you specify a byte adjustment value that represents the difference between the CPE protocol overhead and B-RAS protocol overhead.

The system rounds up the byte adjustment value to the nearest multiple of 4. For example, a value of 6 is rounded to 8, and a value of –10 is rounded to –8.

You do not need to configure a byte adjustment value to account for the downstream ATM network. However, you can specify the byte value to account for additional encapsulations or decapsulations in the downstream network.

Relationship with Other CoS Features

Enabling the overhead accounting feature affects the resulting shaping rates, guaranteed rate, and excess rate parameters, if they are configured.

The overhead accounting feature also affects the egress shaping overhead feature that you can configure at the chassis level. We recommend that you use the egress shaping-overhead feature to account for the Layer 2 overhead of the outgoing interface, and use the overhead-accounting feature to account for downstream traffic with different encapsulations and cell-based networks.

When both features are configured, the total byte adjustment value is equal to the adjusted value of the overhead-accounting feature plus the value of the egress-shaping-overhead feature. For example, if the configured byte adjustment value is 40, and the router internally adjusts the size of each frame by 8, the adjusted overhead accounting value is 48. That value is added to the egress shaping overhead of 24 for a total byte adjustment value of 72.

Related Documentation

- To configure overhead accounting for static Ethernet interfaces, see Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates on page 966
- To configure overhead accounting for dynamic subscriber access, see *Configuring Dynamic Shaping Parameters to Account for Overhead in Downstream Traffic Rates*
- *Setting Shaping Rate and Overhead Accounting Based on PPPoE Vendor-Specific Tags*

Copyright © 2019, Juniper Networks, Inc.

Scheduler Delay Buffering on MIC and MPC Interfaces

To control congestion at the output stage, you can configure the delay-buffer bandwidth. Scheduler delay-buffer bandwidth provides packet buffer space to absorb burst traffic up to the specified duration of delay. After the specified delay buffer becomes full, packets with 100 percent drop probability are dropped from the head of the buffer.

MIC and MPC interfaces support the following default scheduler delay buffer sizes:

- For delay buffer rates below 1 Gbps, the interfaces support delay buffer capacity for 500 ms of buffering.
- For delay buffer rates of 1 Gbps and faster, the interfaces support delay buffer capacity for 100 ms of buffering.
- All tunnel interfaces configured on MIC and MPC interfaces support delay buffer capacity for 100 ms of buffering.

You can configure an explicit buffer size ranging from 4 KB to 256 MB, depending on the MIC or MPC model. However, MIC and MPC interfaces do not support the large delay buffer size configuration statement **q-pic-large-buffer**

Interfaces hosted on MIC and MPC line cards have a certain granularity in the application of configured delay buffer parameters. In other words, the observed hardware value might not exactly match the user-configured value. Nevertheless, the derived values are as close to the configured values as allowed.

When you configure an explicit buffer size, there are 256 points available and the closest point is chosen. High-priority and medium-priority queues use 64 points, and the low-priority queues uses 128.

Related Documentation

- CoS Features and Limitations on MIC and MPC Interfaces on page 930
- Rate Shaping on MIC and MPC Interfaces on page 954
- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972
- Drop Profiles on MIC and MPC Interfaces on page 989

Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs

Service providers often used tiered services that must provide bandwidth for excess traffic as traffic patterns vary. By default, excess bandwidth between a configured guaranteed rate and shaping rate is shared equally among all queues, which might not be optimal for all subscribers to a service.

To manage excess bandwidth:

1. Configure the parameters for the interface.
   a. Configure the shaping rate.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service traffic-control-profiles profile-name]
user@host# set shaping-rate (percent percentage | rate) <burst-size bytes>
```

TIP: On MIC and MPC interfaces, the guaranteed rate and the shaping rate share the value specified for the burst size. If the guaranteed rate has a burst size specified, it is used for the shaping rate; if the shaping rate has a burst size specified, it is used for the guaranteed rate. If you have specified a burst for both rates, the system uses the lesser of the two values.

b. Configure the excess rate.

You can configure an excess rate for all priorities of traffic.

```
[edit class-of-service traffic-control-profiles profile-name]
user@host# set excess-rate (percent percentage | proportion value)
```

Optionally, you can configure an excess rate specifically for high- and low-priority traffic. When you configure the **excess-rate** statement for an interface, you cannot also configure the **excess-rate-low** and **excess-rate-high** statements.

```
[edit class-of-service traffic-control-profiles profile-name]
user@host# set excess-rate-high (percent percentage | proportion value)
user@host# set excess-rate-low (percent percentage | proportion value)
```

BEST PRACTICE: We recommend that you configure either a percentage or a proportion of the excess bandwidth for all schedulers with the same parent in the hierarchy. For example, if you configure interface 1.1 with twenty percent of the excess bandwidth, configure interface 1.2 with eighty percent of the excess bandwidth.

2. (Optional) Configure parameters for the queue.

a. Configure the shaping rate.

```
[edit class-of-service scheduler scheduler-name]
user@host#set shaping-rate (rate | $junos-cos-scheduler-shaping-rate) <burst-size
  bytes>
```

b. Configure the excess rate.

```
[edit class-of-service scheduler scheduler-name]
user@host#set excess-rate (percent percentage | proportion value)
```

c. (Optional) Configure the priority of excess bandwidth for the queue.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service scheduler scheduler-name]
user@host#set excess-priority (low | medium-low | medium-high | high | none)
```

TIP:

For queues, you cannot configure the excess rate in these cases:

- When the **transmit-rate exact** statement is configured. In this case, the shaping rate is equal to the transmit rate and the queue does not operate in the excess region.
- When the scheduling priority is configured as **strict-high**. In this case, the queue gets all available bandwidth and never operates in the excess region.

By default, when traffic exceeds the shaping or guaranteed rates, the system demotes traffic configured with guaranteed high (GH) priority and guaranteed medium (GM) priority. To disable priority demotion, specify the **none** option. You cannot configure this option for queues configured with **transmit-rate** expressed as a percent and when the parent's guaranteed rate is set to zero.

For example, the following statements establish a traffic control profile with a shaping rate of 80 Mbps and an excess rate of 100 percent.

```
[edit class-of-service traffic-control-profiles]
tcp-example-excess {
   shaping-rate 80m;
   excess-rate percent 100;
}
```

The following statements establish a scheduler with an excess rate of 5 percent and a low priority for excess traffic.

```
[edit class-of-service scheduler]
example-scheduler {
   excess-priority low;
   excess-rate percent 5;
}
```

Related Documentation

- Excess Bandwidth Distribution on MIC and MPC Interfaces Overview on page 983
- For more information on hierarchical scheduling and operational modes, see Configuring Hierarchical Schedulers for CoS on page 342.

Copyright © 2019, Juniper Networks, Inc.

Drop Profiles on MIC and MPC Interfaces

This topic covers the following Information

- Drop Profiles on Enhanced Queuing MIC and MPC Interfaces on page 989
- Implicit Scaling of WRED Profiles on page 989

Drop Profiles on Enhanced Queuing MIC and MPC Interfaces

Enhanced queuing (EQ) interfaces on MICs and MPCs support drop profiles as follows:

- Up to 255 drop profiles
- Up to 128 tail-drop priorities for guaranteed low (GL) priorities
- Up to 64 tail-drop priorities for guaranteed high and medium priorities

Implicit Scaling of WRED Profiles

You can oversubscribe scheduler delay buffers by configuring more delay-buffer memory than the system can support. If you oversubscribe the scheduler delay buffers for MIC and MPC interfaces, the system implicitly scales down the configured weighted random early detection (WRED) profiles so that packets are dropped more aggressively from the relatively full queues. This automatic adjustment creates buffer space for packets in the relatively empty queues and provides a sense of fairness among the delay buffers.

Related Documentation

- CoS Features and Limitations on MIC and MPC Interfaces on page 930
- Rate Shaping on MIC and MPC Interfaces on page 954
- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972
- Scheduler Delay Buffering on MIC and MPC Interfaces on page 986
- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355
- Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 381

Copyright © 2019, Juniper Networks, Inc.

Intelligent Oversubscription on MIC and MPC Interfaces Overview

On the MIC and MPC interfaces on MX Series routers, as on other types of interface hardware, arriving packets are assigned to one of two preconfigured traffic classes (network control and best effort) based on their header types and destination media access control (MAC) address. Oversubscription, the situation when the incoming packet rate is much higher than the Packet Forwarding Engine and system can handle, can cause key packets to be dropped and result in a flurry of resends, making the problem worse. However, MIC and MPC interfaces handle oversubscription more intelligently and drops lower priority packets when oversubscription occurs. Protocols such as routing protocols are classified as network control. Protocols such as telnet, FTP, and SSH are classified as best effort. No configuration is necessary.

The following frames and packets are assigned to the network control traffic class:

- ARPs: Ethertype **0x0806** for ARP and **0x8035** for dynamic RARP
- IEEE 802.3ad Link Aggregation Control Protocol (LACP): Ethertype **0x8809** and **0x01** or **0x02** (subtype) in first data byte
- IEEE 802.1ah: Ethertype **0x8809** and subtype **0x03**
- IEEE 802.1g: Destination MAC address **0x01–80–C2–00–00–02** with Logical Link Control (LLC) **0xAAAA03** and Ethertype **0x08902**
- PVST: Destination MAC address **0x01–00–0C–CC–CC–CD** with LLC **0xAAAA03** and Ethertype **0x010B**
- xSTP: Destination MAC address **0x01–80–C2–00–00–00** with LLC **0x424203**
- GVRP: Destination MAC address **0x01–80–C2–00–00–21** with LLC **0x424203**
- GMRP: Destination MAC address **0x01–80–C2–00–00–20** with LLC **0x424203**
- IEEE 802.1x: Destination MAC address **0x01–80–C2–00–00–03** with LLC **0x424203**
- Any per-port **my-mac** destination MAC address
- Any configured global Integrated Bridging and Routing (IRB) **my-mac** destination MAC address

In addition, the following Layer 3 control protocols are assigned to the network control traffic class:

- IGMP query and report: Ethertype **0x0800** and carrying an IPv4 protocol or IPv6 next header field set to 2 (IGMP)
- IGMP DVMRP: IGMP field version = **1** and type = **3**
- IPv4 ICMP: Ethertype **0x0800** and IPv4 protocols = **1** (ICMP)
- IPv6 ICMP: Ethertype **0x86DD** and IPv6 next header field = **0x3A** (ICMP)
- IPv4 or IPv6 OSPF: Ethertype **0x0800** and IPv4 protocol field or IPv6 next header field = **89** (OSPF)

Copyright © 2019, Juniper Networks, Inc.

- IPv4 or IPv6 VRRP: IPv4 Ethertype **0x0800** or IPv6 Ethertype **0x86DD** and IPv4 protocol field or IPv6 next header field = **112** (VRRP)
- IPv4 or IPv6 RSVP: IPv4 Ethertype **0x0800** or IPv6 Ethertype **0x86DD** and IPv4 protocol field or IPv6 next header field = **46** or **134**
- IPv4 or IPv6 PIM: IPv4 Ethertype **0x0800** or IPv6 Ethertype **0x86DD** and IPv4 protocol field or IPv6 next header field = **103**
- IPv4 or IPv6 IS-IS: IPv4 Ethertype **0x0800** or IPv6 Ethertype **0x86DD** and IPv4 protocol field or IPv6 next header field = **124**
- IPv4 router alert: IPv4 Ethertype **0x0800** and IPv4 option field = **0x94** (router alert)

Also, the following Layer 4 control protocols are assigned to the network control traffic class:

- IPv4 and IPv6 BGP: IPv4 Ethertype **0x0800** or IPv6 Ethertype **0x86DD**, TCP port = **179**, and carrying an IPv4 protocol or IPv6 next header field set to 6 (TCP)
- IPv4 and IPv6 LDP: IPv4 Ethertype **0x0800** or IPv6 Ethertype **0x86DD**, TCP or UDP port = **646**, and carrying an IPv4 protocol or IPv6 next header field set to 6 (TCP) or 17 (UDP)
- IPv4 UDP/L2TP control frames: IPv4 Ethertype **0x0800**, UDP port = **1701**, and carrying an IPv4 protocol field set to 17 (UDP)
- DHCP: Ethertype **0x0800**, IPv4 protocol field set to 17 (UDP), and UDP destination port = **0x43** (DHCP service) or **0x44** (DHCP host)
- IPv4 or IPv6 UDP/BFD: Ethertype **0x0800**, UDP port = **3784**, and IPv4 protocol field or IPv6 next header field set to 17 (UDP)

Finally, any PPP encapsulation (Ethertype **0x8863** (PPPoE Discovery) or **0x8864** (PPPOE Session Control)) is assigned to the network control traffic class (queue 3).

NOTE: **These classifications are preconfigured.**

Jitter Reduction in Hierarchical CoS Queues

- Queue Jitter as a Function of the Maximum Number of Queues on page 991
- Default Maximum Queues for Hierarchical Queuing MICs and MPCs on page 992
- Shaping Rate Granularity as a Function of the Rate Wheel Update Period on page 993

Queue Jitter as a Function of the Maximum Number of Queues

Each queuing chip on a Modular Interface Card (MIC) or Modular Port Concentrator (MPC) internally hosts a *rate wheel thread* that updates the *shaper credits* into the *shapers* available at each level of scheduling hierarchy. At each hierarchy level, the length of this update period determines two key characteristics of scheduling:

- The minimum buffer needed for the queue to pass packets without dropping.

Copyright © 2019, Juniper Networks, Inc.

- The degree of jitter encountered in the queue.

At each hierarchy level, the length of the rate wheel update period is dependent upon the number of entities enabled for that node level. Because traffic is queued at Level 5 (queues) and scheduled upwards to Level 1 (the port), the number of entities (queues) enabled at Level 5 determines the number of entities (logical interfaces, interface-sets, or ports) enabled at the other levels of the scheduling hierarchy. By extension, the number of queues enabled for a given scheduler node hierarchy determines the length of the update period at all hierarchy levels. Consequently, limiting the maximum number of queues supported by a hierarchical queuing MIC or MPC can reduce jitter in the queues. To configure the maximum number of queues allowed per hierarchical queuing MIC or MPC, include the max-queues statement at the **[edit chassis fpc *slot-number*]** hierarchy level.

Default Maximum Queues for Hierarchical Queuing MICs and MPCs

The QX chip on a MIC or MPC consists of two symmetrical halves, and each half supports a maximum of 64 K queues (128 K queues per QX chip). The 2-port and 4-port 10-Gigabit Ethernet MICs with XFP and the MPC1_Q line cards have one chipset and can support a maximum of 128 K queues, distributed across the two partitions of the single QX chip. The MPC2 Q and MPC2 EQ line cards have two chipsets and can support a maximum of 256 K queues, distributed across the four partitions of the two QX chips.

Table 145 on page 992 lists the maximum number of queues supported by default and the corresponding rate wheel update period for each hierarchical queuing MIC or MPC.

Table 145: Default Maximum Queues and Corresponding Rate Wheel Update Periods

Router Model	Hierarchical Queuing MIC or MPC	Maximum Queues	Rate Wheel Update Period
MX5, MX10, MX40, and MX80 modular	**2-port 10-Gigabit Ethernet MIC with XFP** The chassis base board hosts one chipset-based Packet Forwarding Engine process that operates in standalone mode. The single QX chip is composed of two partitions that each support 64 K queues for egress ports.	128 K	1.6 ms
MX240, MX480, MX960, MX2010, and MX2020	**MPC1 Q** The MPC1 Q line card hosts one chipset-based Packet Forwarding Engine process that operates in fabric mode. The single QX chip is composed of two partitions that each support 64 K queues for egress ports.	128 K	1.6 ms
	MPC2 Q The MPC2 Q line card hosts two chipset-based Packet Forwarding Engine processes that operate in fabric mode. The two QX chips are composed of four partitions that each support 64 K queues for egress ports.	256 K	1.6 ms
	MPC2 EQ The MPC2 EQ line card hosts two chipset-based Packet Forwarding Engine processes that operate in fabric mode. The two QX chips are composed of four partitions that each support 64 K queues for egress ports.	256 K	2.6 ms

Copyright © 2019, Juniper Networks, Inc.

You can configure hierarchical queuing MICs and MPCs to support a reduced maximum number of queues. Doing so reduces the rate wheel update period used by the QX chip, which in turn reduces jitter in the queues for the egress interfaces hosted on the line card.

Shaping Rate Granularity as a Function of the Rate Wheel Update Period

Reducing the length of the QX chip rate wheel update period, in addition to reducing jitter in the hierarchical scheduling queues, also indirectly increases the shaping granularity.

For a given port line rate and scheduling hierarchy level, the shaping granularity is a function of the minimum shaper credit size and the rate wheel update period in effect as a result of the number of queues supported by the line card.

```
shaping granularity = minimum shaper credit size / rate wheel update period
```

Table 146 on page 993 shows how shaping granularity is calculated for non-enhanced hierarchical queuing MIC and MPC line cards with default values for minimum shaper credit size and for rate wheel update period.

Table 146: Default Shaping Granularities on Non-Enhanced Queuing MICs and MPCs

Port Type	Hierarchy Level	Non-Enhanced Queuing MIC or MPC Defaults		Calculation of Shaping Granularity
		Minimum Credit	Update Period	
1 Gbps Queuing	Level 1 (port), Level 4 (queues)	4 bytes = 32 bits	13.33 ms = 0.01333 sec	32 bits / 0.01333 sec = 2.4 Kbps
	Level 2, Level 3	16 bytes = 128 bits	1.66 ms = 0.00166 sec	128 bits / 0.01333 sec = 9.6 Kbps
10 Gbps Queuing	Level 1 (port), Level 4 (queues)	16 bytes = 128 bits	13.33 ms = 0.01333 sec	128 bits / 0.01333 sec = 9.6 Kbps
	Level 2, Level 3	64 bytes = 512 bits	1.66 ms = 0.00166 sec	512 bits / 0.01333 sec = 38.4 Kbps

Related Documentation

- Example: Reducing Jitter in Hierarchical CoS Queues on page 993
- Per-Unit Queuing and Hierarchical Queuing for MIC and MPC Interfaces on page 978
- Understanding Hierarchical Scheduling for MIC and MPC Interfaces on page 972

Example: Reducing Jitter in Hierarchical CoS Queues

This example shows how to reduce jitter in the output queues for VLAN ports hosted on a hierarchical queuing MPC.

- Requirements on page 994
- Overview on page 994
- Configuration on page 994

Copyright © 2019, Juniper Networks, Inc.

Requirements

This example uses the following Juniper Networks hardware and Junos OS software:

- MX960 router in an IPv4 network and running Junos OS Release 13.2 or later.
- Available Gigabit Ethernet port hosted on FPC slot 2, PIC slot 0, port 0.
- Available Gigabit Ethernet port hosted on port 0 of a Gigabit Ethernet Modular Interface Card (MIC) in PIC slot 0 of an MPC2 Q Modular Port Concentrator (MPC) in FPC slot 5.

Before you begin configuring this example, make sure that the maximum number of queues allowed for the hierarchical queuing MPC in slot 5 has not yet been configured. When you enter the **show chassis fpc 5** command from configuration mode, the **max-queues** statement should not display.

Overview

In this example you configure hierarchical scheduling on a VLAN port hosted on a hierarchical queuing MPC. To reduce jitter in the queues for all egress ports hosted on the MPC, reduce the maximum number of queues allowed for MPC.

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.

```
set interfaces xe-2/0/0 per-unit-scheduler
set interfaces xe-2/0/0 flexible-vlan-tagging
set interfaces xe-2/0/0 unit 0 vlan-id 1
set interfaces xe-2/0/0 unit 0 family inet address 10.1.1.1/24
set interfaces xe-2/0/0 unit * classifiers ieee-802.1 ieee_jitter
set interfaces xe-5/0/0 per-unit-scheduler
set interfaces xe-5/0/0 flexible-vlan-tagging
set interfaces xe-5/0/0 unit 0 vlan-id 1
set interfaces xe-5/0/0 unit 0 family inet address 10.2.1.1/24
set class-of-service-interfaces xe-5/0/0 unit * output-traffic-control-profile tcp
set class-of-service forwarding-classes queue 0 be
set class-of-service forwarding-classes queue 1 ef
set class-of-service forwarding-classes queue 2 af
set class-of-service forwarding-classes queue 3 nc
set class-of-service schedulers be_sch priority low
set class-of-service schedulers ef_sch priority low
set class-of-service schedulers af_sch priority strict-high
set class-of-service schedulers nc_sch priority low
set class-of-service classifiers ieee_jitter forwarding-class be loss-priority low code-points
  000
set class-of-service classifiers ieee_jitter forwarding-class ef loss-priority low code-points
  001
set class-of-service classifiers ieee_jitter forwarding-class af loss-priority low code-points
  010
```


Copyright © 2019, Juniper Networks, Inc.

```
set class-of-service classifiers ieee_jitter forwarding-class nc loss-priority low code-points
  011
set class-of-service scheduler-maps smap_jitter forwarding-class be scheduler be_sch
set class-of-service scheduler-maps smap_jitter forwarding-class ef scheduler ef_sch
set class-of-service scheduler-maps smap_jitter forwarding-class af scheduler af_sch
set class-of-service scheduler-maps smap_jitter forwarding-class nc scheduler nc_sch
set class-of-service traffic-control-profiles tcp scheduler-map smap_jitter
set class-of-service traffic-control-profiles tcp shaping-rate 6g
```

Baseline Configuration

Step-by-Step Procedure

Configure hierarchical scheduling at **xe-5.0.0**.

1. To configure the VLAN 1 input and output at **xe-2/0/0.0** and **xe-5/0/0.0**:

```
[edit]
user@host# set interfaces xe-2/0/0 per-unit-scheduler
user@host# set interfaces xe-2/0/0 flexible-vlan-tagging
user@host# set interfaces xe-2/0/0 unit 0 vlan-id 1
user@host# set interfaces xe-2/0/0 unit 0 family inet address 10.1.1.1/24

user@host# set interfaces xe-5/0/0 per-unit-scheduler
user@host# set interfaces xe-5/0/0 flexible-vlan-tagging
user@host# set interfaces xe-5/0/0 unit 0 vlan-id 1
user@host# set interfaces xe-5/0/0 unit 0 family inet address 10.2.1.1/24
```

2. Map each of four queues to a forwarding class.

```
[edit]
user@host# set class-of-service forwarding-classes queue 0 be
user@host# set class-of-service forwarding-classes queue 1 ef
user@host# set class-of-service forwarding-classes queue 2 af
user@host# set class-of-service forwarding-classes queue 3 nc
```

3. Assign a packet-scheduling priority value to each forwarding class.

```
[edit]
user@host# set class-of-service schedulers be_sch priority low
user@host# set class-of-service schedulers ef_sch priority low
user@host# set class-of-service schedulers af_sch priority strict-high
user@host# set class-of-service schedulers ef_sch priority low
```

4. Customize the default IEEE 802.1p classifier (BA classifier based on Layer 2 header) by defining different values for iEEE 802.1p code points.

```
[edit]
user@host# set class-of-service classifiers ieee_jitter forwarding-class be
  loss-priority low code-points 000
```

Copyright © 2019, Juniper Networks, Inc.

```
user@host# set class-of-service classifiers ieee_jitter forwarding-class ef loss-priority
  low code-points 001
user@host# set class-of-service classifiers ieee_jitter forwarding-class af loss-priority
  low code-points 010
user@host# set class-of-service classifiers ieee_jitter forwarding-class nc
  loss-priority low code-points 011
```

5. Apply the BA classifier to the input of the logical units on **xe-2/0/0**.

```
[edit]
user@host# set interfaces xe-2/0/0 unit * classifiers ieee-802.1 ieee_jitter
```

6. Configure the scheduler map **smap_jitter** to map the forwarding classes to the schedulers.

```
[edit]
user@host# set class-of-service scheduler-maps smap_jitter forwarding-class be
  scheduler be_sch
user@host# set class-of-service scheduler-maps smap_jitter forwarding-class ef
  scheduler ef_sch
user@host# set class-of-service scheduler-maps smap_jitter forwarding-class af
  scheduler af_sch
user@host# set class-of-service scheduler-maps smap_jitter forwarding-class nc
  scheduler nc_sch
```

7. Configure the traffic control profile **tcp** to combine the scheduler map **smap_jitter** (that maps the forwarding classes to the schedulers for port-based scheduling) with a shaping rate (for hierarchical scheduling).

```
[edit]
user@host# set class-of-service traffic-control-profiles tcp scheduler-map
  smap_jitter
user@host# set class-of-service traffic-control-profiles tcp shaping-rate 6g
```

8. Apply the traffic control profile to the router output at **xe-5/0/0**.

```
[edit]
user@host# set class-of-service-interfaces xe-5/0/0 unit *
  output-traffic-control-profile tcp
```

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

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


Copyright © 2019, Juniper Networks, Inc.

Results Confirm your configuration by entering **show interfaces** and **show cloass-of-service** commands from configuration mode. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@host# show interfaces
xe-2/0/0 {
   per-unit-scheduler;
   flexible-vlan-tagging;
   unit 0 {
      vlan-id 1;
      family inet {
         address 10.1.1.1/24;
      }
   }
}
xe-5/0/0 {
   per-unit-scheduler;
   flexible-vlan-tagging;
   unit 0 {
      vlan-id 1;
      family inet {
         address 10.2.1.1/24;
      }
   }
}
```

```
[edit]
user@host# show class-of-service
classifiers {
   ieee-802.1 ieee_jitter {
      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-classes {
   queue 0 be;
   queue 1 ef;
   queue 2 af;
   queue 3 nc;
}
traffic-control-profiles {
   tcp {
      scheduler-map smap_jitter;
```

Copyright © 2019, Juniper Networks, Inc.

```
      shaping-rate 6g;
    }
  }
  interfaces {
    xe-2/0/0 {
      unit * {
        classifiers {
          ieee-802.1 ieee_jitter;
        }
      }
    }
    xe-5/0/0 {
      unit * {
        output-traffic-control-profile tcp;
      }
    }
  }
  scheduler-maps {
    smap_jitter {
      forwarding-class be scheduler be_sch;
      forwarding-class ef scheduler ef_sch;
      forwarding-class af scheduler af_sch;
      forwarding-class nc scheduler nc_sch;
    }
  }
  schedulers {
    be_sch {
      priority low;
    }
    ef_sch {
      priority low;
    }
    af_sch {
      priority strict-high;
    }
    nc_sch {
      priority low;
    }
  }
```

Verification

Confirm that the configuration is working properly

- Measuring End-to-End Jitter to Establish the Baseline on page 998
- Configuring Jitter Reduction on page 999
- Measuring End-to-End Jitter to Verify Jitter Reduction on page 999

Measuring End-to-End Jitter to Establish the Baseline

Purpose Establish a baseline measurement by noting the amount of jitter that occurs when the hierarchical queuing line card hosting the egress port is configured with the default maximum number of queues.

Copyright © 2019, Juniper Networks, Inc.

Action To measure jitter:

1. Pass traffic through the VLAN.
2. Measure the variation in packet delay for selected packets in the data flow.

Configuring Jitter Reduction

Purpose Reduce jitter in the VLAN port output queues.

Action

1. Configure a reduced maximum number of queues for egress ports on the hierarchical queuing MPC in slot 5, thereby reducing the jitter in the port queues.

```
[edit]
user@host# set chassis fpc 5 max-queue 64k
```

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

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

Measuring End-to-End Jitter to Verify Jitter Reduction

Purpose Measure the amount of jitter that occurs when the hierarchical queuing line card hosting the egress port is configured with a reduced maximum number of queues.

Action To measure jitter:

1. Pass traffic through the VLAN.
2. Measure the variation in packet delay for selected packets in the data flow.

Related Documentation

- Jitter Reduction in Hierarchical CoS Queues on page 991
- max-queues on page 1206

CoS on Ethernet Pseudowires in Universal Edge Networks Overview

You can apply rewrite rules and classifiers to an Ethernet pseudowire on MIC and MPC interfaces on MX Series routers. In an edge network, the pseudowire can represent a single customer.

To create the pseudowires, you use logical tunnel (LT) interfaces that connect two virtual routing forwarding (VRF) instances. To provide CoS to the LT interface, you can apply

Copyright © 2019, Juniper Networks, Inc.

classifiers and rewrite rules. Rewrite rules enable you to rewrite packet header information by specifying various CoS values, including DiffServ code point (DSCP) and IP precedence.

NOTE: **Scheduling is not supported on LT interfaces in the current release.**

For example, a VPLS instance is connected to a Layer 3 routing instance. The logical tunnel labeled **lt-9/0/0.0** is configured with **vpls** as the family, and **lt-9/0/0.1** is configured with **inet** as the family. You can apply a rewrite rule and classifier for DSCP to **lt-9/0/0.1**, which can represent a business subscriber.

Related Documentation

- Configuring CoS on an Ethernet Pseudowire for Multiservice Edge Networks on page 1000

CoS Scheduling Policy on Logical Tunnel Interfaces Overview

You can configure a CoS scheduling policy on a logical tunnel interface (LT ifl). Logical tunnel interfaces can be used to terminate a pseudowire into a virtual routing and forwarding (VRF) instance. If an LT device is used to terminate a pseudowire, CoS scheduling policies can be applied on the LT interface to manage traffic entering the pseudowire. You accomplish this by configuring the hierarchical-scheduler attribute on the physical interface.

This feature is supported on MIC and MPC interfaces.

Related Documentation

- Configuring a CoS Scheduling Policy on Logical Tunnel Interfaces on page 977
- Configuring Hierarchical Schedulers for CoS on page 342
- CoS on Ethernet Pseudowires in Universal Edge Networks Overview on page 999
- Configuring CoS on an Ethernet Pseudowire for Multiservice Edge Networks on page 1000

Configuring CoS on an Ethernet Pseudowire for Multiservice Edge Networks

You can configure rewrite rules and classifiers to logical tunnel (LT) interfaces that are configured to represent Ethernet pseudowires.

This feature is supported on MIC and MPC interfaces.

To configure CoS on an LT interface configured for an Ethernet pseudowire:

1. Configure a pair of LT interfaces to represent a pseudowire.

   To apply rewrite rules and classifiers to the pseudowire, you must assign one of the LT interfaces to the **inet** family.

```
[edit]
user@host#edit interfaces lt-fpc/pic/port
user@host#edit unit logical-unit-number
user@host#set encapsulation encapsulation
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host#set family (inet | inet6 | iso | mpls)
user@host#set peer-unit unit-number
```

2. Configure the rewrite rule.

   The available rewrite rule types for an LT interface are **dscp** and **inet-precedence**.

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

3. Configure the classifier.

   The available classifier types for an LT interface are **dscp** and **inet-precedence**.

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

4. Apply the rewrite rule and classifier to the LT interface that you assigned to the **inet** family.

```
[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 on page 999
- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388.
- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36.

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 1002
- Hardware Requirements for Inline Services on the LNS on page 1002

Copyright © 2019, Juniper Networks, Inc.

Guidelines for Applying CoS to the LNS

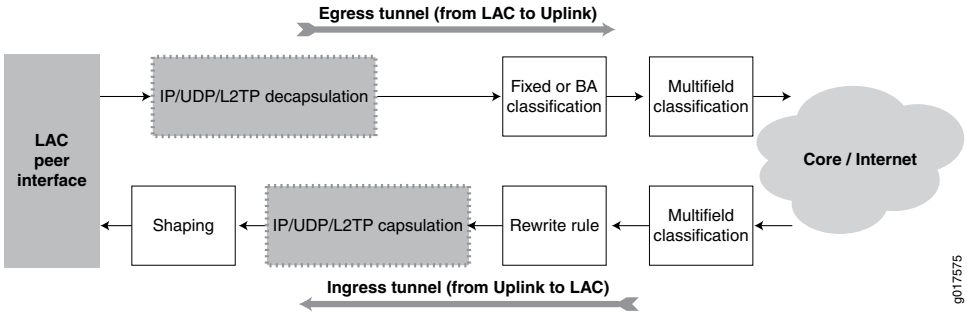
In L2TP configurations, IP, UDP, and L2TP headers are added to packets arriving at a PPP subscriber interface on the L2TP access concentrator (LAC) before being tunneled to the LNS.

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 72 on page 1002 shows the classifier and rewrite rules that you can configure on an LNS inline service.

Figure 72: 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 147 on page 1002 lists the supported inline services and peer interfaces for each MIC and MPC combination.

Table 147: Hardware Requirements for L2TP LNS Inline Services

MPC Module	Inline Service Support–With Per-Session Shaping	Inline Service Support–Without Per-Session Shaping
MPC2E-3D-NG	No	Yes
MPC2E-3D-NG-Q MX80	Yes	Yes
MPC-3D-16XGE-SFPP	No	No

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring Static CoS for an L2TP LNS Inline Service on page 1003
- *Configuring Dynamic CoS for an L2TP LNS Inline Service*

Configuring Static 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 static CoS for an L2TP LNS inline service:

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 si-fpc/port/pic  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:

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces si-fpc/port/pic 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 class-of-service interfaces si-fpc/port/pic 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 class-of-service interfaces si-fpc/port/pic unit logical-unit-number
  rewrite-rules]
user@host# set dscp(rewrite-name | <default>) protocol protocol-types
user@host# set dscp-ipv6 (rewrite-name | <default>)
```

   - To apply the rewrite rule for the ToS value:

```
[edit class-of-service interfaces si-fpc/port/pic unit logical-unit-number
  rewrite-rules]
user@host# set inet-precedence (rewrite-name | <default>) protocol
  protocol-types
```

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 class-of-service traffic-control-profiles profile-name]
user@host# set overhead-accounting (frame-mode | cell-mode) <bytes byte-value
```

6. Apply the traffic-control profile.

```
[edit class-of-service interfaces si-fpc/port/pic unit logical-unit-number]
user@host# set output-traffic-control-profile profile-name
```

BEST PRACTICE: To limit bandwidth for tunneled sessions with default CoS configurations, we recommend that you also configure CoS for the remaining traffic on the static service interface.

Copyright © 2019, Juniper Networks, Inc.

```
[edit class-of-service interfaces si-fpc/port/pic ]
user@host# set output-traffic-control-profile-remaining profile-name
```

Related Documentation

- CoS for L2TP LNS Inline Services Overview on page 1001
- Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates on page 966

CoS on Circuit Emulation ATM MICs Overview

The following class-of-service features are supported on Circuit Emulation ATM MICs:

- Traffic shaping and scheduling—Traffic shaping determines the maximum amount of traffic that can be transmitted on an interface.

  You can configure three different categories of ATM service: constant bit rate (**cbr**), non-real-time variable bit rate (**nrvbr**), and real-time variable bit rate (**rtvbr**). The service category works in conjunction with ATM cell parameters **peak-rate**, **sustained-rate**, and **max burst-size** to impose traffic shaping, transmit rate, shaping rate, and default excess rate for an ATM queue.

  Beginning with Junos OS Release 14.2, for an ATM MIC with SFP (MIC-3D-8OC3-2OC12-ATM), you can configure CBR bandwidth speeds up to 622 Mbps (OC12 speed) per virtual circuit (VC). To enable each VC on an ATM MIC with SFP to support up to line rate traffic in OC12 mode, define both of the following parameters in the traffic shaping and scheduling profile:

  - Configure **cbr** as the ATM service category (**atm-service**) .
  - Configure up to a maximum of 1,412,829 cells per second (cps) as the ATM peak cell rate (**peak-rate**) .

  To set the OC12 per-VC port speed for an ATM MIC with SFP, configure the **oc12-stm4** port speed option for the desired port.

  NOTE: When you configure up to OC12 CBR bandwidth speed per VC on an ATM MIC with SFP, the actual Layer 3 payload throughput you obtain depends on the ATM encapsulation type and IP packet size that you use. The Layer 3 payload rate is the maximum Layer 3 (IP) payload throughput achieved for a given Layer 2 traffic rate.

- Policing—Policing, or rate limiting, enables you to limit the amount of traffic that passes into or out of the interface. It works with firewall filters to thwart denial-of-service (DoS) attacks.

  Networks police traffic by limiting the input or output transmission rate of a class of traffic on the basis of user-defined criteria. The ATM policer controls the maximum rate of traffic sent from or received on the interface on which it is applied.

Copyright © 2019, Juniper Networks, Inc.

To apply limits to the traffic flow, configure the **cdvt** and **peak-rate** parameters within the policer. Define the **policing-action** parameter as **discard**, **discard-tag**, and **count** to set a consequence for the packets that exceed these limits. The consequence is usually a higher loss priority so that if the packets encounter downstream congestion, they are discarded first.

Release History Table

Release	Description
14.2	Beginning with Junos OS Release 14.2, for an ATM MIC with SFP (MIC-3D-8OC3-2OC12-ATM), you can configure CBR bandwidth speeds up to 622 Mbps (OC12 speed) per virtual circuit (VC).

Related Documentation

- Configuring CoS on Circuit Emulation ATM MICs on page 1006
- *Configuring Port Speed on Multi-Rate MICs*

Configuring CoS on Circuit Emulation ATM MICs

On MX Series routers, you can configure the following class-of-service features on Circuit Emulation ATM MICs:

- Traffic shaping and scheduling—Traffic shaping determines the maximum amount of traffic that can be transmitted on an interface.
- Policing—Policing, or rate limiting, enables you to limit the amount of traffic that passes into or out of the interface. It works with firewall filters to thwart denial-of-service (DoS) attacks.

To configure a traffic shaping and scheduling profile on a Circuit Emulation ATM MIC:

1. Configure the traffic shaping and scheduling profile and specify the service category that determines the traffic shaping parameter for the ATM queue at the ATM MIC.

```
[edit class-of-service traffic-control-profile traffic-control-profile-name]
user@host# set atm-service (cbr | nrtvbr| rtvbr)
```

NOTE: Beginning with Junos OS Release 14.2, to configure up to OC12 constant bit rate (CBR) bandwidth speed per virtual circuit (VC) on an ATM MIC with SFP (MIC-3D-8OC3-2OC12-ATM), specify **cbr** as the ATM service category.

2. Configure the transmit rate, shaping rate, and default excess rate for the ATM queue.

```
[edit class-of-service traffic-control-profile traffic-control-profile-name]
user@host# set peak-rate peak-rate
user@host# set sustained-rate rate
```


Copyright © 2019, Juniper Networks, Inc.

```
user@host# set max-burst-size cells
```

NOTE: Beginning with Junos OS Release 14.2, to configure up to OC12 CBR bandwidth speed per VC on an ATM MIC with SFP, specify up to 1,412,829 cells per second (cps) as the ATM peak cell rate.

To configure an ATM policer for a Circuit Emulation ATM MIC:

1. Create a policer for each cell in the ATM packet. A policer defines the maximum traffic that can flow through an interface and further determines the actions to be taken when the traffic exceeds the defined limits.

   ```
   [edit firewall]
   user@host# set atm-policer atm-policer-name
   ```

2. Define the policer parameters. Configure the **atm-service** option. Apply limits to the traffic flow by configuring the **cdvt** and **peak-rate** parameters within the policer and define the **policing-action** parameter to set a consequence for the packets that exceed the traffic limits.

   ```
   [edit firewall atm-policer atm-policer-name]
   user@host# set logical-interface-policer
   user@host# set atm-service (cbr | rtvbr | nrtvbr | ubr)
   user@host# set cdvt rate
   user@host# set peak-rate rate
   user@host# set policing-action (discard | discard-tag | count)
   ```

3. Apply the traffic-shaping profile at the class-of-service interface level.

   ```
   [edit class-of-service interfaces at-fpc/pic/port unit unit-number]
   user@host# set output-traffic-control-profile traffic-control-profile-name
   ```

4. Apply the policer at the interface level.

   ```
   [edit interfaces at-fpc/pic/port unit unit-number]
   user@host# set atm-policer input-atm-policer atm-policer-name
   ```

You can verify the configuration by using the **show class-of-service traffic-control-profile** ***traffic-control-profile-name*** command.

Copyright © 2019, Juniper Networks, Inc.

Release History Table

Release	Description
14.2	Beginning with Junos OS Release 14.2, to configure up to OC12 constant bit rate (CBR) bandwidth speed per virtual circuit (VC) on an ATM MIC with SFP (MIC-3D-8OC3-2OC12-ATM), specify **cbr** as the ATM service category.
14.2	Beginning with Junos OS Release 14.2, to configure up to OC12 CBR bandwidth speed per VC on an ATM MIC with SFP, specify up to 1,412,829 cells per second (cps) as the ATM peak cell rate.

Related Documentation

- *ATM Support on Circuit Emulation PICs Overview*
- CoS on Circuit Emulation ATM MICs Overview on page 1005

Understanding IEEE 802.1p Inheritance push and swap from a Transparent Tag

M320 router interfaces and MX Series router interfaces on Modular Interface Cards (MIC) or Modular Port Concentrators (MPCs) support configurable IEEE 802.1p inheritance of push and swap bits from the transparent tag of each incoming packet which allows you to classify incoming packets based on the IEEE 802.1p bits from the transparent tag.

During a tagging operation, Junos OS by default inherits the IEEE 802.1p bits from incoming tags in swap and push operations from the known tags configured on the interface.

It can be useful to override the default behavior by configuring Junos OS to inherit the IEEE 802.1p bits from a transparent tag, and to classify incoming packets based on the IEEE 802.1p bits of the incoming transparent tag. The configuration statements **swap-by-poppush** and **transparent** enable Junos OS to do this.

By default, during a swap operation, the IEEE 802.1p bits of the VLAN tag remain unchanged. When the **swap-by-poppush** operation is enabled on a logical interface, the swap operation is treated as a **pop** operation followed by **push** operation. The **pop** operation removes the existing tag and the associated IEEE 802.1p bits and the push operation copies the inner VLAN IEEE 802.1p bits to the IEEE bits of the VLAN or VLANs being pushed. As a result, the IEEE 802.1p bits are inherited from the incoming transparent tag.

To classify incoming packets based on the IEEE 802.1p bits from the transparent tag, include the **transparent** statement at the **[edit class-of-service interfaces** *interface-name* **unit** *logical-unit-number* **classifiers ieee-802.1 vlan-tag]** hierarchy level.

To configure Junos OS to inherit the IEEE 802.1p bits from the transparent tag, include the **swap-by-poppush** statement at the **[edit interfaces** ***interface-name*** **unit** ***logical-unit-number*****]** hierarchy level.

NOTE: IEEE 802.1p Inheritance push and swap is only supported on untagged and single-tagged logical interfaces, and is not supported on dual-tagged logical interfaces.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- *swap-by-poppush*
- transparent on page 1313
- *Understanding swap-by-poppush*
- Configuring IEEE 802.1p Inheritance push and swap from the Transparent Tag on page 1009
- *Understanding Transparent Tag Operations and IEEE 802.1p Inheritance*

Configuring IEEE 802.1p Inheritance push and swap from the Transparent Tag

To classify incoming packets based on the IEEE 802.1p bits from the transparent tag, include the **transparent** statement at the **[edit class-of-service interfaces** *interface-name* **unit** *logical-unit-number* **classifiers ieee-802.1 vlan-tag]** hierarchy level.

Tagged Interface Example

The following example configuration specifies the classification based on the transparent VLAN tag.

```
edit
class-of-service {
  interfaces {
    ge-3/0/1 {
      unit 0 {
        classifiers {
          ieee-802.1 default vlan-tag transparent;
        }
      }
    }
  }
}
```

To configure Junos OS to inherit the IEEE 802.1p bits from the transparent tag, include the **swap-by-poppush** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

The following is a configuration to swap and push VLAN tags and allow inheritance of the IEEE 802.1p value from the transparent VLAN tag in incoming packets.

```
edit
  ge-3/0/0 {
  vlan-tagging;
  encapsulation vlan-ccc;
  unit 0 {
    encapsulation vlan-ccc;
    vlan-id 100;
    swap-by-poppush;
    input-vlan-map {
      swap-push;
      tag-protocol-id 0x9100;
      inner-tag-protocol-id 0x9100;
      vlan-id 500;
```

Copyright © 2019, Juniper Networks, Inc.

```
        inner-vlan-id 400;
      }
      output-vlan-map {
        pop-swap;
        inner-vlan-id 100;
        inner-tag-protocol-id 0x88a8;
      }
    }
}
```

The **swap-by-poppush** statement causes a swap operation to be done as a pop followed by a push operation. So for the outer tag, the incoming S-Tag is popped and a new tag is pushed. As a result, the S-Tag inherits the IEEE 802.1p bits from the transparent tag. The inner tag is then pushed, which results in the inner tag inheriting the IEEE 802.1p bits from the transparent tag.

Untagged Interface Example

The following is a configuration to push two VLAN tags and allow inheritance of the IEEE 802.1p value from the transparent VLAN tag in the incoming packet.

```
[edit]
ge-3/0/1 {
  encapsulation ccc;
  unit 0 {
    input-vlan-map {
      push-push;
      tag-protocol-id 0x9100;
      inner-tag-protocol-id 0x9100;
      vlan-id 500;
      inner-vlan-id 400;
    }
    output-vlan-map{
      pop-pop;
    }
  }
}
```

No additional configuration is required to inherit the IEEE 802.1p value, as the **push** operation inherits the IEEE 802.1p values by default.

The following configuration specifies the classification based on the transparent VLAN tag.

```
[edit]
class-of-service {
  interfaces {
    ge-3/0/1 {
      unit 0 {
        classifiers {
          ieee-802.1 default vlan-tag transparent;
        }
      }
    }
  }
```


Copyright © 2019, Juniper Networks, Inc.

```
}
```

Related Documentation

- transparent on page 1313
- *swap-by-poppush*
- Understanding IEEE 802.1p Inheritance push and swap from a Transparent Tag on page 1008
- *Understanding swap-by-poppush*
- *Understanding Transparent Tag Operations and IEEE 802.1p Inheritance*

CoS on Application Services Modular Line Card Overview

The Application Services Modular Line Card (AS MLC) is designed to run services for real-time traffic on MX240, MX480, and MX960 routers. It consists of three main components:

- Application Services Modular Carrier Card (AS MCC)
- Application Services Modular Processing Card (AS MXC)
- Application Services Modular Storage Card (AS MSC)

It supports class-of-service (CoS) features to ensure the quality of service (QoS) for real-time traffic that is sensitive to latency on a network. The AS MLC supports the following CoS features:

- Code-point Aliases—A code-point alias assigns a name to a pattern of code-point bits. On the AS MLC, you can use the code-point alias name for CoS components such as classifiers and drop-profile maps.
- Classification—Packet classification refers to the examination of ingress packets. On the AS MLC, the traffic flowing from the Modular Processing Card (AS MXC) towards the Modular Carrier Card (AS MCC) supports three types of classification:
  - Behavior Aggregate (BA)—BA classifier can be configured on the aggregated logical interfaces to classify traffic flowing from the AS MXC towards the AS MCC. With BA classification you can set the forwarding class and loss priority of a packet based on its code points. The AS MLC only supports IP classification (classification based on Type of Service (ToS) and Differentiated Services Code Point (DSCP)) and classification is supported for the IPv4 family only. The Media Flow Controller application sets appropriate DSCP/ToS code-point in the packet that is evaluated by the BA classifier on the AS MCC to classify the packet.
  - Multifield Classification—With multifield classifiers you can set the class and loss priority based on one or more of the following packet header fields: destination address, destination port, DSCP, IP protocol, and source address. Multifield classifiers are used when a simple BA classifier is insufficient to classify a packet.

Copyright © 2019, Juniper Networks, Inc.

  - Fixed Classification—Fixed classification can be configured on logical interfaces by specifying a forwarding class to be applied to all packets received by the logical interface, regardless of the packet contents.

- Scheduling—Schedulers enable you to define the buffer sizes, delay buffer size, drop profile map, excess priority, excess rate percentage, output-traffic-control profile, priority, scheduler-map, shaping rate, transmit rate, and random early detection (RED) drop profiles to be applied to a particular queue for packet transmission.

The AS MLC provides CoS features in the following deployment scenarios:

- HTTP Reverse Proxy— In HTTP reverse proxy configurations, the service provider provides services to a set of domains (content providers) that buy content caching capability from the service provider. Clients connect to content providers through virtual IP (VIP) addresses. Service providers in the reverse proxy scenario generally deploy the routers with AS MLC hardware to honor service requests (such as caching) from the domain users.
- HTTP Transparent Proxy—In HTTP transparent proxy configurations, the service provider implements the AS MLC to improve its own caching capability and reduce the load on its own network. Implementing caching on an MX Series router with an AS MLC improves the retrieval speeds for data and optimizes the back-end network utilization.
- Mixed Mode—In mixed mode both reverse proxy and transparent proxy are configured on the same router.

CoS Implementation in HTTP Reverse Proxy Scenario

In the reverse proxy configuration, the AS MXC provides content to multiple domains. The Media Flow Controller application on an AS MXC implements the differentiated services by setting the DSCP or IP precedence value for the IP packets traversing from the AS MXC to an AS MCC on the AS MLC hardware. The Modular Carrier Card uses these values to classify the packet and provide a suitable level of service.

The Media Flow Controller application detects the domain it serves and marks the DSCP values or the IP precedence bit value based on how important the traffic corresponding to that particular domain is. The service provider operator also sets a behavior aggregate (BA) classifier on the aggregated interfaces on the AS MCC. Based on the DSCP/IP precedence bits, the classifier sets the forwarding class and packet loss priority for the packet. The forwarding class and the packet loss priority values govern the next-hop behavior of the packet traversing the Juniper Networks router.

Unlike a firewall, the Media Flow Controller application implemented on the AS MLC hardware marks the DSCP/IP precedence values based on the application layer protocols. This feature ensures that important traffic flowing from the AS MXC gets a higher priority and is processed accordingly. For example, if MPEG is implemented on the egress, the drop precedence for each frame can be different such that the P and B frames (which require more processing) are dropped before the I frames, resulting in a better quality video for the end user.

Copyright © 2019, Juniper Networks, Inc.

For traffic received on the ingress interfaces, end-to-end quality-of-service (QoS) policies ensure that the traffic arriving at the interface has the right DSCP values and the traffic is prioritized based on the forwarding class and packet loss priority values.

CoS Implementation in Transparent Proxy Scenario

In the HTTP transparent proxy configuration, the service provider deploys the AS MLC hardware to reduce its own traffic instead of serving a particular domain. The Media Flow Controller application marks the DSCP bits based on its own requirements rather than those of the domains. Besides this, the CoS implementation for the egress interface is similar to the reverse proxy configuration scenario. The incoming packets follow the QoS policies applied at the WAN interface.

CoS Implementation in Mixed-Mode Scenario

In mixed mode both reverse proxy and transparent proxy configuration coexist on the same AS MLC hardware. In such a scenario, reverse proxy is configured on an aggregated interface and transparent proxy is configured on a regular interface with the Media Flow Controller application marking the appropriate DSCP values for both the configurations. The individual CoS implementation in both the scenarios remains similar to the implementation discussed in "CoS Implementation in HTTP Reverse Proxy Scenario" on page 1012 and "CoS Implementation in Transparent Proxy Scenario" on page 1013

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 27

Configuring Class of Service on Aggregated, Channelized, and Gigabit Ethernet Interfaces

- Limitations on CoS for Aggregated Interfaces on page 1015
- Policer Support for Aggregated Ethernet Interfaces Overview on page 1017
- Understanding Schedulers on Aggregated Interfaces on page 1018
- Examples: Configuring CoS on Aggregated Interfaces on page 1019
- Hierarchical Schedulers on Aggregated Ethernet Interfaces Overview on page 1021
- Configuring Hierarchical Schedulers on Aggregated Ethernet Interfaces on page 1022
- Example: Configuring Scheduling Modes on Aggregated Interfaces on page 1023
- Enabling VLAN Shaping and Scheduling on Aggregated Interfaces on page 1029
- Example: Configuring Per-Unit Schedulers for Channelized Interfaces on page 1031
- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 1033

Limitations on CoS for Aggregated Interfaces

Both Ethernet and SONET/SDH interfaces can be aggregated. The limitations covered here apply to both.

There are some restrictions when you configure CoS on aggregated Ethernet and SONET/SDH interfaces:

- Chassis scheduling, described in "Applying Scheduler Maps to Chassis-Level Queues" on page 778, is not supported on aggregated interfaces, because a chassis scheduler applies to the entire PIC and not just to one interface.
- An aggregated interface is a pseudo-interface. Therefore, CoS queues are not associated with the aggregated interface. Instead, CoS queues are associated with the member link interfaces of the aggregated interface.
- When you apply CoS parameters to the aggregated interface, they are applied to the CoS queues of the member link interfaces. You do not, however, apply CoS classifiers and rewrite rules directly to the member link interfaces.
- You cannot apply a scheduler map to a member link of an aggregate interface.

Copyright © 2019, Juniper Networks, Inc.

- Rate-based CoS components such as scheduler, shaper, and policer are not supported on mixed rate aggregated Ethernet links. However, the default CoS settings are supported by default on the mixed rate aggregated Ethernet links.
- Ingress queuing is not supported on aggregated interfaces.

When the scheduler map of the aggregate interface has schedulers configured for absolute transmit rate, the scheduler for the member link interfaces is scaled to the speed of each member link interface. Each member link interface has an automatic scheduler map that is not visible in the CLI. This scheduler map is allocated when the member link is added to the aggregate interface and is deleted when the member link is removed from the aggregate interface.

- If you configure the scheduler transmit rate of the aggregate interface as an absolute rate, the software uses the following formula to scale the transmit rate of each member link:

  ```
  transmit rate of member link interface =
  (configured transmit rate of aggregate interface /
  total speed of aggregate interface) *
  (total speed of member link interface / total configured percent) * 100
  ```

- If you configure the scheduler transmit rate of the aggregate interface as a percentage, the software uses the following formula to scale the transmit rate of each member link:

  ```
  transmit rate percent of member link interface =
  (configured transmit rate percent of aggregate interface /
  total configured percent) * 100
  ```

The total configured percent is the sum of the configured transmit rate of all schedulers in terms of percentage of the total speed of the aggregate interface.

- All the other parameters for the schedulers, including priority, drop profile, and buffer size, are copied without change from the scheduler of the aggregated interface to the member link interfaces.
- The configuration related to the logical interfaces, including classifiers and rewrite rules, is copied from the aggregated logical interface configuration to the member link logical interfaces.
- For the scheduler map applied to an aggregated interface, if you configure a transmission rate in absolute terms, then the traffic of all the member link interfaces might be affected if any of the member link interfaces go up or down.

When applying CoS configurations to bundles, you must apply the CoS configuration directly to the bundle, not to the physical ports that are part of the bundle. The device may give you a false commit if you apply a CoS configuration directly to a physical port that is part of a bundle. This limitation applies if you attempt to configure a physical port that is already a member of a bundle or if you attempt to add a physical port to a bundle that already has a CoS configuration applied to it.

Copyright © 2019, Juniper Networks, Inc.

If you want to add a physical port to a bundle that already has a CoS configuration, you must:

1. Remove the CoS configuration from the port.
2. Commit your changes on the device.
3. Add the port to the bundle. The CoS configurations that are present on the bundle will be applied to the port you are adding to the bundle.
4. Commit you changes on the device.

In addition, if you want to remove a physical port from a bundle and ensure the physical port has the appropriate CoS configurations, you must:

1. Remove the port from the bundle.
2. Commit your changes on the device.
3. Apply the applicable CoS configuration to the port.
4. Commit your changes on the device.

Related Documentation

- Examples: Configuring CoS on Aggregated Interfaces on page 1019

Policer Support for Aggregated Ethernet Interfaces Overview

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

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

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

NOTE: This feature is supported on the following platforms: T Series routers (excluding T4000 Type 5 FPCs), M120, M10i, M7i (CFEB-E only), M320 (SFPC only), MX240, MX480, and MX960 with DPC, MIC, and MPC interfaces, and EX Series switches.

The following usage scenarios are supported:

Copyright © 2019, Juniper Networks, Inc.

- Interface policers used by the following configuration:

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

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

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

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

The following usage scenarios are not supported:

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

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

Related Documentation

- shared-bandwidth-policer on page 1284

Understanding Schedulers on Aggregated Interfaces

You can apply a class-of-service (CoS) configuration to aggregated Ethernet and aggregated SONET/SDH interfaces. The CoS configuration applies to all member links included in the aggregated interface. You cannot apply different CoS configurations to the individual member links.

You can configure shaping for aggregated Ethernet interfaces that use interfaces originating from Gigabit Ethernet IQ2 PICs. However, you cannot enable shaping on aggregated Ethernet interfaces when there is a mixture of ports from Intelligent Queuing (IQ) and Intelligent Queuing 2 (IQ2) PICs in the same bundle.

You cannot configure a shaping rate and guaranteed rate on an aggregated Ethernet interface with member interfaces on IQ or IQ2 PICs. The commit will fail. These statements are allowed only when the member interfaces are Enhanced Queuing DPC Gigabit Ethernet interfaces.

To view the summation of the queue statistics for the member links of an aggregate interface, issue the **show interfaces queue** command. To view the queue statistics for each member link, issue the **show interfaces queue *aggregated-interface-name*** command.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring Schedulers on page 258
- Configuring Scheduler Maps on page 258
- Applying Scheduler Maps Overview on page 259

Examples: Configuring CoS on Aggregated Interfaces

This example illustrates how CoS scheduler parameters are configured and applied to aggregated interfaces.

Applying Scaling Formula to Absolute Rates

Configure queues as follows when the total speed of member link interfaces is 100 Mbps (the available bandwidth is 100 Mbps):

```
[edit class-of-service]
schedulers {
  be {
    transmit-rate 10m;
  }
  af {
    transmit-rate 20m;
  }
  ef {
    transmit-rate 80m;
  }
  nc {
    transmit-rate 30m;
  }
}
```

The total configured transmit rates of the aggregated interface is **10m** + **20m** + **80m** + **30m** = 140 Mbps, meaning the transmit rate is overconfigured by 40 percent. Therefore, the software scales down the configuration to match the 100 Mbps of available bandwidth, as follows:

```
be = (10/140) * 100 = 7 percent of 100 Mbps = 7 Mbps
af = (20/140) * 100 = 14 percent of 100 Mbps = 14 Mbps
ef = (80/140) * 100 = 57 percent of 100 Mbps = 57 Mbps
nc = (30/140) * 100 = 21 percent of 100 Mbps = 21 Mbps
```

Applying Scaling Formula to Mixture of Percent and Absolute Rates

Configure the following mixture of percent and absolute rates:

```
[edit class-of-service]
schedulers {
  be {
    transmit-rate 20 percent;
  }
  af {
    transmit-rate 40 percent;
  }
  ef {
    transmit-rate 150m;
```

Copyright © 2019, Juniper Networks, Inc.

```
    }
    nc {
      transmit-rate 10 percent;
    }
}
```

Assuming 300 Mbps of available bandwidth, the configured percentages correlate with the following absolute rates:

```
schedulers {
    be {
      transmit-rate 60m;
    }
    af {
      transmit-rate 120m;
    }
    ef {
      transmit-rate 150m;
    }
    nc {
      transmit-rate 30m;
    }
}
```

The software scales the bandwidth allocation as follows:

```
be = (60/360) * 100 = 17 percent of 300 Mbps = 51 Mbps
af = (120/360) * 100 = 33 percent of 300 Mbps = 99 Mbps
ef = (150/360) * 100 = 42 percent of 300 Mbps = 126 Mbps
nc = (30/360) * 100 = 8 percent of 300 Mbps = 24 Mbps
```

Configuring an Aggregated Ethernet Interface

Configure an aggregated Ethernet interface with the following scheduler map:

```
[edit class-of-service]
scheduler-maps {
    aggregated-sched {
      forwarding-class be scheduler be;
      forwarding-class af scheduler af;
      forwarding-class ef scheduler ef;
      forwarding-class nc scheduler nc;
    }
}
schedulers {
    be {
      transmit-rate percent 10;
      buffer-size percent 25;
    }
    af {
      transmit-rate percent 20;
      buffer-size percent 25;
    }
    ef {
      transmit-rate 80m;
```


Copyright © 2019, Juniper Networks, Inc.

```
    buffer-size percent 25;
  }
  nc {
    transmit-rate percent 30;
    buffer-size percent 25;
  }
}
```

In this case, the transmission rate for the member link scheduler map is as follows:

- **be**—7 percent
- **af**—14 percent
- **ef**—57 percent
- **nc**—21 percent

If you add a Fast Ethernet interface to the aggregate, the aggregate bandwidth is 200 Mbps, and the transmission rate for the member link scheduler map is as follows:

- **be**—10 percent
- **af**—20 percent
- **ef**—40 percent
- **nc**—30 percent

Related Documentation

- Limitations on CoS for Aggregated Interfaces on page 1015

Hierarchical Schedulers on Aggregated Ethernet Interfaces Overview

On MX Series routers, you can apply hierarchical schedulers on aggregated ethernet bundles using interface sets. This feature enables you to configure a group of virtual LANs (VLANs) and control their bandwidth. This feature is supported at egress only.

You can configure interface sets for aggregated Ethernet (AE) interfaces created under static configurations. You can configure class-of-service parameters on AE interfaces, in either link-protect or non-link-protect mode. You can configure these parameters at the AE physical interface level. The CoS configuration is fully replicated for all AE member links in link-protect mode. You can control the way these parameters are applied to member links in non-link-protect mode by configuring the AE interface to operate in scaled mode or replicate mode.

The link membership list and scheduler mode of the interface set is 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 scheduler mode is configured to replicate member links, the scheduling parameters of the interface set are copied to each of the member links.

Copyright © 2019, Juniper Networks, Inc.

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 (scaling factor is 1/A where A is the number of active links in the bundle) and applied to each of the AE interface member links.

To configure an interface set, include the **interface-set** statement at the **[edit class-of-service interfaces]** hierarchy level.

To apply scheduling and queuing parameters to the interface set, include the **output-traffic-control-profile** *profile-name* statement at the **[edit class-of-service interfaces** *interface-name* **interface-set** *interface-set-name***]** hierarchy level.

To apply an output traffic scheduling and shaping profile for the remaining traffic to the logical interface or interface set, include the **output-traffic-control-profile-remaining** *profile-name* statement at the **[edit class-of-service interfaces** *interface-name***]** hierarchy level or the **[edit class-of-service interfaces** *interface-name* **interface-set** *interface-set-name***]** hierarchy level.

Related Documentation

- Configuring Hierarchical Schedulers on Aggregated Ethernet Interfaces on page 1022
- output-traffic-control-profile-remaining on page 1221
- Controlling Remaining Traffic on page 295

Configuring Hierarchical Schedulers on Aggregated Ethernet Interfaces

The following example shows the creation of an interface set for aggregated Ethernet interfaces in a static Ethernet configuration.

To configure interface sets for aggregated Ethernet (AE) interfaces created under static configurations:

1. Create the AE interfaces.

```
[edit]
user@host# show chassis | display set
set chassis aggregated-devices ethernet device-count 10
```

2. Configure the AE physical interfaces and member links.

   user@host# show interfaces | display set

```
set interfaces ge-5/2/0 gigether-options 802.3ad ae0
set interfaces ge-5/2/1 gigether-options 802.3ad ae0
set interfaces ae0 hierarchical-scheduler maximum-hierarchy-levels 2
set interfaces ae0 flexible-vlan-tagging
set interfaces ae0 unit 0 vlan-id 100
set interfaces ae0 unit 1 vlan-id 101
set interfaces ae0 unit 2 vlan-id 102
set interfaces ae0 unit 3 vlan-id 103
set interfaces ae0 unit 4 vlan-id 104
```


Copyright © 2019, Juniper Networks, Inc.

3. Configure the interface set.

```
set interfaces interface-set ifset1-ae0 interface ae0 unit 0
set interfaces interface-set ifset1-ae0 interface ae0 unit 1
```

4. Configure class-of-service parameters for the interface sets.

```
set class-of-service interfaces interface-set ifset1-ae0 output-traffic-control-profile
  tcp
```

NOTE: You also need to configure the parameters of the traffic control profile. For more information, see the Related Documentation section on this page.

5. Configure scheduler mode.

```
set class-of-service interfaces ae0 member-link-scheduler scale
```

Related Documentation

- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- Hierarchical Schedulers on Aggregated Ethernet Interfaces Overview on page 1021
- Example: Configuring Shared Resources on Ethernet IQ2 Interfaces on page 832
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

Example: Configuring Scheduling Modes on Aggregated Interfaces

You can configure class-of-service parameters, such as queuing or shaping parameters on aggregated interfaces, in either link-protect or non-link-protect mode. You can configure these parameters for per-unit schedulers, hierarchical schedulers, or shaping at the physical and logical interface level. You can control the way these parameters are applied by configuring the aggregated interface to operate in **scale** or **replicate** mode.

You can apply these parameters on the following routers:

- MX Series router interfaces on EQ DPCs
- MX Series router interfaces on MICs or MPCs through Junos OS Release 10.2 (non-link-protect mode only)
- M120 or M320 routers
- T Series router interfaces on IQ2 PICs
- PTX Series Packet Transport Routers

You can configure the applied parameters for aggregated interfaces operating in non-link-protected mode. In link-protected mode, only one link in the bundle is active at

Copyright © 2019, Juniper Networks, Inc.

a time (the other link is a backup link) so schedulers cannot be scaled or replicated. In non-link-protected mode, all the links in the bundle are active and send traffic; however, there is no backup link. If a link fails or is added to the bundle in non-link-protected mode, the links' traffic is redistributed among the active links.

To set the scheduling mode for aggregated interfaces, include the **scale** or **replicate** option of the **member-link-scheduler** statement at the **[edit class-of-service interfaces ae*n*]** hierarchy level, where *n* is the configured number of the interface:

```
[edit class-of-service interfaces aen]
member-link-scheduler (replicate | scale);
```

By default, if you do not include the **member-link-scheduler** statement, scheduler parameters are applied to the member links in the **scale** mode (also called "equal division mode").

The aggregated Ethernet interfaces are otherwise configured as usual. For more information on configuring aggregated Ethernet interfaces, see the *Junos OS Network Interfaces Library for Routing Devices*.

The following examples set **scale** mode on the **ae0** interface and **replicate** mode on the **ae1** interface.

```
[edit class-of-service]
interfaces ae0 {
   member-link-scheduler scale;
}

[edit class-of-service]
interfaces ae1 {
   member-link-scheduler replicate;
}
```

NOTE: The **member-link-scheduler** statement only appears for aggregated interfaces. You configure this statement for aggregated interfaces in non-link-protected mode. For more information about link protection modes, see the *Network Interfaces Configuration Guide*.

Aggregated interfaces support both hierarchical and per-unit schedulers.

NOTE: The **traffic-control-profiles** statement is not supported for PTX Series Packet Transport Routers.

Copyright © 2019, Juniper Networks, Inc.

When interface parameters are using the **scale** option of the **member-link-scheduler** statement, the following parameters under the **[edit class-of-service traffic-control-profiles *traffic-control-profile-name*]** configuration are scaled on egress when hierarchical schedulers are configured:

- **shaping-rate** (PIR)
- **guaranteed-rate** (CIR)
- **delay–buffer-rate**

When interface parameters are using the **scale** option of the **member-link-scheduler** statement, the following parameters under the **[edit class-of-service schedulers *scheduler-name*]** configuration are scaled on egress when per-unit schedulers are configured:

- **transmit-rate**
- **buffer-size**

NOTE: You cannot apply a hierarchical scheduler at the interface set level for an ae interface. (Interface sets cannot be configured under an ae interface.)

The following configuration parameters are not supported on **ae** interfaces in non-link-protection mode:

- Input scheduler maps
- Input traffic control profiles
- Input shaping rates

The following configuration conventions are also not supported:

- Scaling of the **input-traffic-control-profile-remaining** statement.
- The **scheduler-map-chassis** statement and the **derived** option for the **ae** interface. Chassis scheduler maps should be applied under the physical interfaces.
- Dynamic and demux interfaces are not supported as part of the **ae** bundle.

Depending on the whether the **scale** or **replicate** option is configured, the **member-link-scheduler** statement operates in either scaled mode (also called “equal division mode”) or replicated mode, respectively.

In scaled mode, a VLAN can have multiple flows that can be sent over multiple member links of the **ae** interface. Likewise, a member link can receive traffic from any VLAN in the **ae** bundle. In scaled mode, the physical interface bandwidth is divided equally among all member links of the **ae** bundle.

Copyright © 2019, Juniper Networks, Inc.

In scaled mode, the following scheduler parameter values are divided equally among the member links:

- When the parameters are configured using traffic control profiles, then the parameters scaled are the shaping rate, guaranteed rate, and delay buffer rate.
- When the parameters are configured using scheduler maps, then the parameters scaled are the transmit rate and buffer size. Shaping rate is also scaled if you configure it in bits per second (bps). Shaping rate is not scaled if you configure it as a percentage of the available interface bandwidth.

For example, consider an **ae** bundle between routers R1 and R2 consisting of three links. These are **ge-0/0/1**, **ge-0/0/2** and **ge-0/0/3** (**ae0**) on R1; and **ge-1/0/0**, **ge-1/0/1**, and **ge-1/0/2** (**ae2**) on R2. Two logical interfaces (units) are also configured on the **ae0** bundle on R1: **ae0.0** and **ae0.1**.

On **ae0**, traffic control profiles on R1 are configured as follows:

- **ae0** (the physical interface level) has a PIR of 450 Mbps.
- **ae0.0** (VLAN 100 at the logical interface level) has a PIR of 150 Mbps and a CIR of 90 Mbps.
- **ae0.1** (VLAN 200 at the logical interface level) has a PIR of 90 Mbps and a CIR of 60 Mbps.

In scaled mode, the **ae0** PIR is first divided among the member physical interfaces. Because there are three members, each receives 450 / 3 = 150 Mbps as a derived value. So the scaled PIR for the members interfaces is 150 Mbps each.

However, there are also two logical interfaces (**ae0.0** and **ae0.1**) and VLANs (100 and 200) on **ae0**. Traffic can leave on any of the three physical interfaces (**ge-0/0/1**, **ge-0/0/2**, or **ge-0/0/3**) in the bundle. Therefore, two derived logical interfaces are added to the member links to represent the two VLANs.

There are now six logical interfaces on the physical interfaces of the links making up the **ae** bundle, one set for VLAN 100 and the other for VLAN 200:

- **ge-0/0/1.0** and **ge-0/0/1.1**
- **ge-0/0/2.0** and **ge-0/0/2.1**
- **ge-0/0/3.0** and **ge-0/0/3.1**

The traffic control profile parameters configured on **ae0.0** are divided across all the underlying logical interfaces (the unit 0s). In the same way, the traffic control profile parameters configured on **ae0.1** are divided across all the underlying logical interfaces (the unit 1s).

Therefore, the derived values of the scaled parameters on the interfaces are:

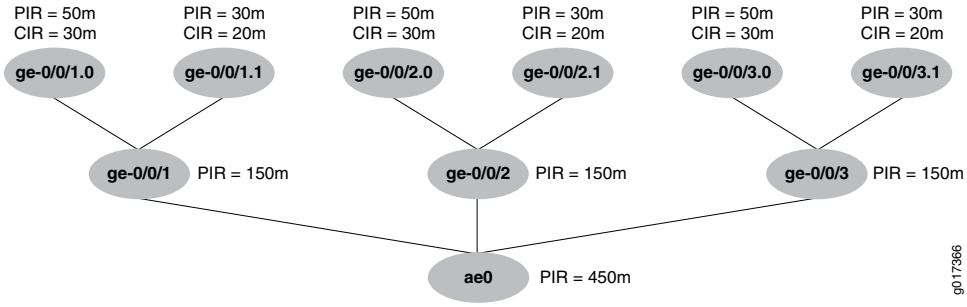
- For **ge-0/0/1.0** and **ge-0/0/2.0** and **ge-0/0/3.0**, each CIR = 90 / 3 = 30 Mbps, and each PIR = 150 / 3 = 50 Mbps.

Copyright © 2019, Juniper Networks, Inc.

- For **ge-0/0/1.1** and **ge-0/0/2.1** and **ge-0/0/3.1**, each CIR = 60 / 3 = 20 Mbps, and each PIR = 90 / 3 = 30 Mbps.

The scaled values are shown in Figure 73 on page 1027.

Figure 73: Scaled Mode for Aggregated Ethernet Interfaces



In scaled mode, when a new member link is added to the bundle, or an existing member link is either removed or fails, then the scaling factor (based on the number of active links) is recomputed and the new scheduler or traffic control profile parameters are reassigned. Only the PIR, CIR, and buffer parameters are recomputed: all other parameters are simply copied at each level.

NOTE: In **show class-of-service scheduler-map** commands, values derived in scaled mode instead of explicitly configured are flagged with &**sf**_n_ suffix, where *n* indicates the value of the scaling factor.

The following sample shows the output for the scheduler map named **smap-all-abs** with and without a scaling factor:

```
user@host> show class-of-service scheduler-map
Scheduler map: smap-all-abs, Index: 65452

Scheduler: q0_sch_abs, Forwarding class: be, Index: 6775
Transmit rate: 40000000 bps, Rate Limit: none, Buffer size: remainder,
Priority: low
    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>
```

```
user@host> show class-of-service scheduler-map
Scheduler map: smap-all-abs, Index: 65452

Scheduler: q0_sch_abs&**sf**3, Forwarding class: be, Index: 2128
Transmit rate: 13333333 bps, Rate Limit: none, Buffer size: remainder,
Priority: low
    Excess Priority: unspecified
    Drop profiles:
```

Copyright © 2019, Juniper Networks, Inc.

```
    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>
```

NOTE: There can be multiple scheduler maps created with different scaling factors, depending on when the child interfaces come up. For example, if there are only two active children on a parent interface, a new scheduler map with a scaling factor of 2 is created. The scheduler map name is **smap-all-abs&**sf**2**.

In replicated mode, in contrast to scaled mode, the configured scheduler parameters are simply replicated, not divided, among all member links of the **ae** bundle.

In replicated mode, the following scheduler parameter values are replicated among the member links and logical interfaces:

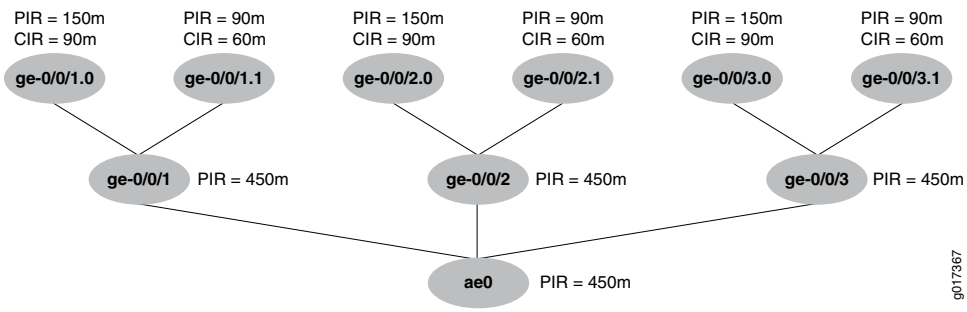
- When the parameters are configured using traffic control profiles, then the parameters replicated are the shaping rate, guaranteed rate, and delay buffer rate.
- When the parameters are configured using scheduler maps, then the parameters replicated are the transmit rate and buffer size.

If the scheduler parameters in the example configuration between routers R1 and R2 are applied with the **member-link-scheduler replicate** statement and option, the following parameters are applied:

- The **ae0** PIR is copied among the member physical interfaces. Each receives 450 Mbps as a PIR.
- For each logical interface unit **.0**, the configured PIR and CIR for **ae0.0** is replicated (copied). Each logical interface unit **.0** receives a PIR of 150 Mbps and a CIR of 90 Mbps.
- For each logical interface unit **.1**, the configured PIR and CIR for **ae0.1** is replicated (copied). Each logical interface unit **.1** receives a PIR of 90 Mbps and a CIR of 60 Mbps.

The replicated values are shown in Figure 74 on page 1028.

Figure 74: Replicated Mode for Aggregated Ethernet Interfaces



Copyright © 2019, Juniper Networks, Inc.

In replicated mode, when a new member link is added to the bundle, or an existing member link is either removed or fails, the values are either copied or deleted from the required levels.

Related Documentation

- How Schedulers Define Output Queue Properties on page 254
- Default Schedulers Overview on page 257
- Configuring Schedulers on page 258

Enabling VLAN Shaping and Scheduling on Aggregated Interfaces

Virtual LAN (VLAN) shaping (per-unit scheduling) is supported on aggregated Ethernet interfaces when link protection is enabled on the aggregated Ethernet interface. When VLAN shaping is configured on aggregate Ethernet interfaces with link protection enabled, the shaping is applied to the active child link.

To configure link protection on aggregated Ethernet interfaces, include the **link-protection** statement at the **[edit interfaces ae*x* aggregated-ether-options]** hierarchy level.

Traffic passes only through the designated primary link. This includes transit traffic and locally generated traffic on the router. When the primary link fails, traffic is routed through the backup link. You also can reverse traffic, from the designated backup link to the designated primary link. To revert back to sending traffic to the primary designated link when traffic is passing through the designated backup link, use the **revert** command. For example, **request interfaces revert ae0**.

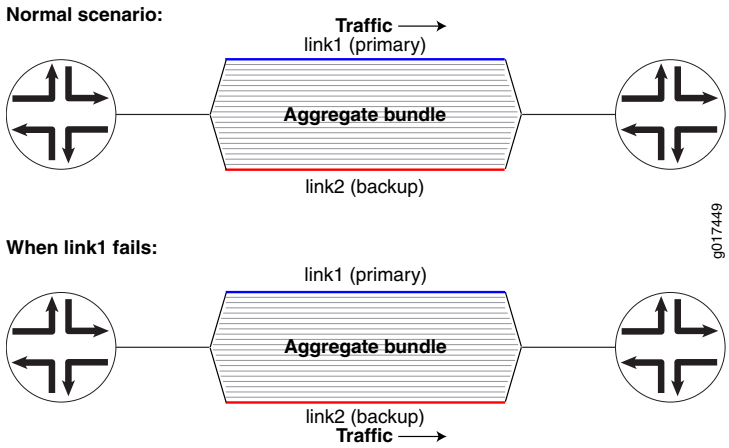
To configure a primary and a backup link, include the **primary** and **backup** statements at the **[edit interfaces ge-*fpc/pic/port* gigether-options 802.3ad ae*x*]** hierarchy level or the **[edit interfaces xe-*fpc/pic/port* fastether-options 802.3ad ae*x*]** hierarchy level.

To disable link protection, delete the **link-protection** statement at **[edit interfaces ae*x* aggregated-ether-options link-protection]** hierarchy level. To display the active, primary, and backup link for an aggregated Ethernet interface, use the operational mode command **show interfaces redundancy ae*x***.

Figure 75 on page 1030 shows how the flow of traffic changes from primary to backup when the primary link in an aggregate bundle fails.

Copyright © 2019, Juniper Networks, Inc.

Figure 75: Aggregated Ethernet Primary and Backup Links



This example configures two Gigabit Ethernet interfaces (**primary** and **backup**) as an aggregated Ethernet bundle (**ae0**) and enables link protection so that a shaping rate can be applied to VLANs on the aggregated Ethernet interface.

```
[edit class-of-service]
interface ae0 {
   shaping-rate 300m;
}
[edit interfaces]
ge-1/0/0 {
   gigether-options {
      802.3ad ae0 primary;
   }
}
ge-1/0/1 {
   gigether-options {
      802.3ad ae0 backup;
   }
}
ae0 {
   aggregated-ether-options {
      lacp {
         periodic slow;
      }
      link-protection {
         enable;
      }
   }
}
```

Related Documentation

- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300
- Example: Applying Scheduling and Shaping to VLANs on page 312

Copyright © 2019, Juniper Networks, Inc.

Example: Configuring Per-Unit Schedulers for Channelized Interfaces

You can configure per-unit scheduling on T1 and DS0 physical interfaces configured on channelized DS3 and STM1 IQ PICs. To enable per-unit scheduling, configure the **per-unit-scheduler** statements at the **[edit interfaces *interface-name*]** hierarchy level.

When per-unit scheduling is enabled on the channelized PICs, you can associate a scheduler map with the physical interface. For more information about configuring scheduler maps, see "Configuring Scheduler Maps" on page 258.

NOTE: If you configure the per-unit-scheduler statement on the physical interface of a 4-port channelized OC-12 IQ PIC and configure 975 logical interfaces or data link connection identifiers (DLCIs), some of the logical interfaces or DLCIs will drop all packets intermittently.

The following example configures per-unit scheduling on a channelized DS3 PIC and an STM1 IQ PIC.

```
[edit interfaces]
ct3-5/3/1 {
   partition 1 interface-type t1;
}
t1-5/3/1:1 {
   per-unit-scheduler; # This enables per-unit scheduling
   encapsulation frame-relay;
   unit 0 {
      dlci 1;
      family inet {
         address 10.0.0.2/32;
      }
   }
}
ct3-5/3/0 {
   partition 1 interface-type ct1;
}
ct1-5/3/0:1 {
   partition 1 timeslots 1 interface-type ds;
}
ds-5/3/0:1:1 {
   per-unit-scheduler; # This enables per-unit scheduling
   encapsulation frame-relay;
   unit 0 {
      dlci 1;
      family inet {
         address 10.0.0.1/32;
      }
   }
}
cau4-3/0/0 {
   partition 1 interface-type ce1;
}
```

Copyright © 2019, Juniper Networks, Inc.

```
cstm1-3/0/0 {
   no-partition 1 interface-type cau4;
}
ce1-3/0/0:1 {
   partition 1 timeslots 1 interface-type ds;
}
ds-3/0/0:1:1 {
   per-unit-scheduler; # This enables per-unit scheduling
   encapsulation frame-relay;
   unit 0 {
      dlci 1;
      family inet {
         address 10.1.1.1/32;
      }
   }
}

[edit class-of-service]
classifiers {
   dscp all-traffic-dscp {
      forwarding-class assured-forwarding {
         loss-priority low code-points 001010;
      }
      forwarding-class expedited-forwarding {
         loss-priority low code-points 101110;
      }
      forwarding-class best-effort {
         loss-priority low code-points 101010;
      }
      forwarding-class network-control {
         loss-priority low code-points 000110;
      }
   }
}
forwarding-classes {
   queue 0 best-effort;
   queue 1 assured-forwarding;
   queue 2 expedited-forwarding;
   queue 3 network-control;
}
interfaces {
   ds-3/0/0:1:1 {
      unit 0 {
         scheduler-map schedule-mlppp;
      }
   }
   ds-5/3/0:1:1 {
      unit 0 {
         scheduler-map schedule-mlppp;
      }
   }
   t1-5/3/1:1 {
      unit 0 {
         scheduler-map schedule-mlppp;
```


Copyright © 2019, Juniper Networks, Inc.

```
    }
  }
}
scheduler-maps {
  schedule-mlppp {
    forwarding-class expedited-forwarding scheduler expedited-forwarding;
    forwarding-class assured-forwarding scheduler assured-forwarding;
    forwarding-class best-effort scheduler best-effort;
    forwarding-class network-control scheduler network-control;
  }
}
schedulers {
  best-effort {
    transmit-rate percent 2;
    buffer-size percent 5;
    priority low;
  }
  assured-forwarding {
    transmit-rate percent 7;
    buffer-size percent 30;
    priority low;
  }
  expedited-forwarding {
    transmit-rate percent 90 exact;
    buffer-size percent 60;
    priority high;
  }
  network-control {
    transmit-rate percent 1;
    buffer-size percent 5;
    priority strict-high;
  }
}
```

Related Documentation

- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300
- Example: Applying Scheduler Maps and Shaping Rate to DLCIs on page 308

Applying Layer 2 Policers to Gigabit Ethernet Interfaces

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

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

Copyright © 2019, Juniper Networks, Inc.

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

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

Examples: Applying Layer 2 Policers to a Gigabit Ethernet Interface

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

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

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

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

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

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

Related Documentation

- Controlling Network Access Using Traffic Policing Overview on page 115
- Overview of Tricolor Marking Architecture on page 171

Copyright © 2019, Juniper Networks, Inc.

PART 5

Configuration Statements and Operational Commands

- Configuration Statements on page 1037
- Operational Commands on page 1327

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 28

Configuration Statements

- action on page 1045
- address (CoS on ATM Interfaces) on page 1046
- adjust-minimum on page 1047
- adjust-percent on page 1048
- application-profile on page 1049
- application-sets (Services CoS) on page 1050
- applications (Services CoS) on page 1050
- atm-options on page 1051
- atm-policer on page 1053
- atm-scheduler-map on page 1054
- atm-service on page 1055
- buffer-size (Schedulers) on page 1056
- bypass-queuing-chip on page 1058
- bytes (Dynamic Traffic Shaping) on page 1059
- cbr on page 1060
- cdvt on page 1061
- cell-mode (Dynamic Traffic Shaping) on page 1062
- class (CoS-Based Forwarding) on page 1063
- class (Forwarding Classes) on page 1064
- classification-override on page 1065
- classifiers (Definition) on page 1066
- classifiers (Logical Interface) on page 1067
- classifiers (Physical Interface) on page 1068
- classifiers (Routing Instance) on page 1069
- class-of-service on page 1070
- class-of-service (Protocols MPLS) on page 1071
- code-point-aliases on page 1072
- code-point on page 1073

Copyright © 2019, Juniper Networks, Inc.

- code-points (CoS) on page 1074
- copy-plp-all on page 1075
- copy-tos-to-outer on page 1075
- copy-tos-to-outer-ip-header on page 1076
- copy-tos-to-outer-ip-header-transit on page 1077
- data (FTP) on page 1077
- default (CoS Host Outbound Traffic) on page 1078
- delay-buffer-rate on page 1079
- destination-address (Services CoS) on page 1080
- destination (Interfaces) on page 1081
- discard (Forwarding Class) on page 1082
- drop-probability (Interpolated Value) on page 1083
- drop-probability (Percentage) on page 1084
- drop-profile (Schedulers) on page 1085
- drop-profile-map (Schedulers) on page 1086
- drop-profiles on page 1087
- drop-timeout (Forwarding Class) on page 1088
- dscp (Services CoS) on page 1089
- dscp (CoS Classifiers) on page 1090
- dscp (CoS Interfaces) on page 1091
- dscp (Multifield Classifier) on page 1092
- dscp (Rewrite Rules) on page 1093
- dscp (Rewrite Rules on Physical Interface) on page 1094
- dscp-code-point (CoS Host Outbound Traffic) on page 1095
- dscp-ipv6 (CoS Rewrite Rules) on page 1096
- egress-policer-overhead on page 1097
- egress-shaping-overhead on page 1098
- enhanced (forwarding-class-accounting) on page 1100
- enhanced-priority-mode on page 1103
- epd-threshold on page 1104
- excess-bandwidth-share on page 1105
- excess-priority on page 1106
- excess-rate on page 1107
- excess-rate-medium-high on page 1108
- excess-rate-medium-low on page 1109
- excess-rate-high on page 1110
- excess-rate-low on page 1111

Copyright © 2019, Juniper Networks, Inc.

- exclude-queue-overhead-bytes on page 1112
- exp on page 1113
- exp-push-push-push on page 1114
- exp-swap-push-push on page 1115
- explicit-null-cos on page 1116
- fabric (Class-of-Service) on page 1117
- family (CoS on ATM Interfaces) on page 1118
- family (Multifield Classifier) on page 1119
- fill-level (Drop Profiles) on page 1120
- fill-level (Interpolated Value) on page 1121
- filter (Applying to an Interface) on page 1122
- filter (Applying to a Logical Interface) on page 1123
- filter (Configuring) on page 1124
- firewall on page 1125
- flexible-queuing-mode on page 1126
- flexible-vlan-tagging on page 1127
- forwarding-class (Services PIC Classifiers) on page 1128
- forwarding-class (ATM2 IQ Scheduler Maps) on page 1129
- forwarding-class (BA Classifiers) on page 1130
- forwarding-class (CoS Host Outbound Traffic) on page 1131
- forwarding-class (Forwarding Policy) on page 1132
- forwarding-class (Fragmentation) on page 1133
- forwarding-class (Interfaces) on page 1134
- forwarding-class (Multifield Classifiers) on page 1134
- forwarding-class (Restricted Queues) on page 1135
- forwarding-class-accounting on page 1136
- forwarding-class-default (Forwarding Policy) on page 1137
- forwarding-classes-interface-specific on page 1138
- (OBSOLETE) forwarding-classes (Class-of-Service) on page 1139
- forwarding-policy on page 1140
- fragment-threshold (Forwarding Class Maps) on page 1141
- fragmentation-map on page 1142
- fragmentation-maps on page 1143
- frame-mode (Dynamic Traffic Shaping) on page 1144
- frame-relay-de (Defining Loss Priority Maps) on page 1145
- frame-relay-de (Defining Loss Priority Rewrites) on page 1146
- from (Services CoS) on page 1147

Copyright © 2019, Juniper Networks, Inc.

- ftp (Services CoS) on page 1148
- guaranteed-rate on page 1149
- hierarchical-scheduler on page 1150
- high-plp-max-threshold on page 1151
- high-plp-threshold on page 1152
- host-outbound-traffic (Class-of-Service) on page 1153
- hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 1154
- ieee-802.1 (Classifier on Physical Interface) on page 1156
- ieee-802.1 (Host Outbound Traffic) on page 1157
- ieee-802.1 (Rewrite Rules on Logical Interface) on page 1158
- ieee-802.1 (Rewrite Rules on Physical Interface) on page 1159
- ieee-802.1ad on page 1160
- import (Classifiers) on page 1161
- import (Rewrite Rules) on page 1161
- inet-precedence (Classifier on Physical Interface) on page 1162
- inet-precedence (CoS Classifiers) on page 1163
- inet-precedence (CoS Rewrite Rules) on page 1164
- inet-precedence (Rewrite Rules on Physical Interface) on page 1165
- inet6-precedence (CoS Rewrite Rules) on page 1166
- ingress-policer-overhead on page 1167
- ingress-queuing-filter on page 1169
- ingress-shaping-overhead on page 1170
- input-excess-bandwidth-share on page 1171
- input-policer on page 1172
- input-scheduler-map on page 1173
- input-shaping-rate (Logical Interface) on page 1174
- input-shaping-rate (Physical Interface) on page 1175
- input-three-color on page 1176
- input-traffic-control-profile on page 1177
- input-traffic-control-profile-remaining on page 1178
- interface-set (Ethernet Interfaces) on page 1179
- interface-set (Hierarchical Schedulers) on page 1180
- interfaces (CoS) on page 1181
- internal-node on page 1183
- interpolate on page 1184
- iq-policing-filter on page 1185
- irb on page 1186

Copyright © 2019, Juniper Networks, Inc.

- layer2-policer on page 1187
- linear-red-profile on page 1188
- linear-red-profiles on page 1189
- logical-bandwidth-policer on page 1190
- logical-interface-aggregate-statistics on page 1191
- logical-interface-policer on page 1192
- loss-priority (BA Classifiers) on page 1194
- loss-priority (Firewall Filter) on page 1194
- loss-priority (Rewrite Rules) on page 1195
- loss-priority (Scheduler Drop Profiles) on page 1196
- loss-priority (Simple Firewall Filter) on page 1197
- loss-priority-maps on page 1198
- loss-priority-maps (Assigning to an Interface) on page 1199
- loss-priority-rewrites on page 1200
- loss-priority-rewrites (Assigning to an Interface) on page 1201
- low-plp-max-threshold on page 1202
- low-plp-threshold on page 1203
- lsp-next-hop (CoS-Based Forwarding) on page 1204
- match-direction (Services CoS) on page 1204
- max-burst-size on page 1205
- max-queues on page 1206
- max-queues-per-interface on page 1208
- member-link-scheduler on page 1210
- mode (Layer 2 Tunneling Protocol Shaping) on page 1211
- multilink-class on page 1212
- next-hop (Class-Of-Service) on page 1213
- next-hop-map on page 1214
- no-fragmentation on page 1215
- non-lsp-next-hop on page 1216
- output-forwarding-class-map on page 1216
- output-policer on page 1217
- output-three-color on page 1218
- output-traffic-control-profile on page 1219
- output-traffic-control-profile-remaining on page 1221
- overhead-accounting on page 1222
- packet-timestamp on page 1223
- peak-rate on page 1224

Copyright © 2019, Juniper Networks, Inc.

- per-session-scheduler on page 1225
- per-unit-scheduler on page 1226
- plp-to-clp on page 1227
- policer (Configuring) on page 1228
- policing-action on page 1230
- policy-map on page 1231
- priority (ATM2 IQ Schedulers) on page 1232
- priority (Fabric Priority) on page 1233
- priority (Fabric Queues, Schedulers) on page 1234
- priority (Schedulers) on page 1235
- protocol (Host Outbound Traffic) on page 1236
- protocol (Rewrite Rules) on page 1237
- protocol (Schedulers) on page 1238
- queue (Global Queues) on page 1239
- queue (Restricted Queues) on page 1240
- queue-depth on page 1241
- queue-threshold on page 1242
- red-buffer-occupancy on page 1244
- reflexive | revert | reverse on page 1245
- restricted-queues on page 1246
- rewrite-rules (CoS Host Outbound Traffic) on page 1247
- rewrite-rules (Definition) on page 1248
- rewrite-rules (Interfaces) on page 1249
- rewrite-rules (Physical Interfaces) on page 1251
- routing-instances (CoS) on page 1252
- rtvbr on page 1253
- rule (Services CoS) on page 1255
- rule-set (Services CoS) on page 1256
- scheduler (Fabric Queues) on page 1256
- scheduler (Scheduler Map) on page 1257
- scheduler-map (Fabric Queues) on page 1257
- scheduler-map (Interfaces and Traffic-Control Profiles) on page 1258
- scheduler-map-chassis on page 1259
- scheduler-maps (For ATM2 IQ Interfaces) on page 1260
- scheduler-maps (For Most Interface Types) on page 1261
- schedulers (CoS) on page 1262
- schedulers (Interfaces) on page 1263

Copyright © 2019, Juniper Networks, Inc.

- services (CoS) on page 1263
- shaping on page 1264
- shaping-rate (Applying to an Interface) on page 1265
- shaping-rate (Schedulers) on page 1267
- shaping-rate (Oversubscribing an Interface) on page 1269
- shaping-rate-excess-high on page 1271
- shaping-rate-excess-low on page 1273
- shaping-rate-excess-medium-high on page 1275
- shaping-rate-excess-medium-low on page 1276
- shaping-rate-priority-high on page 1277
- shaping-rate-priority-low on page 1279
- shaping-rate-priority-medium on page 1281
- shaping-rate-priority-medium-low on page 1282
- shaping-rate-priority-strict-high on page 1283
- shared-bandwidth-policer (Configuring) on page 1284
- shared-instance on page 1285
- shared-scheduler on page 1286
- simple-filter (Applying to an Interface) on page 1287
- simple-filter on page 1288
- sip (Application Profile) on page 1289
- source-address (Services CoS) on page 1290
- strict-priority-scheduler on page 1290
- sustained-rate on page 1291
- syslog (Services CoS) on page 1291
- system-defaults on page 1292
- term (Firewall Filter) on page 1293
- term (Services CoS) on page 1295
- term (Simple Filter) on page 1296
- then (Services CoS) on page 1297
- three-color-policer (Applying) on page 1298
- three-color-policer (Configuring) on page 1299
- traffic-class (Tunnels) on page 1301
- traffic-class-map on page 1302
- traffic-class-map (Apply to Interface) on page 1304
- traffic-control-profiles on page 1305
- traffic-manager on page 1306
- translation-table on page 1310

Copyright © 2019, Juniper Networks, Inc.

- transmit-rate (Schedulers) on page 1311
- transmit-weight on page 1313
- transparent on page 1313
- tri-color on page 1314
- tunnel on page 1315
- tunnel-services (Chassis) on page 1316
- unit on page 1318
- vbr on page 1319
- vc-cos-mode on page 1320
- vci on page 1321
- video (Application Profile) on page 1322
- vlan-tag on page 1323
- vlan-tagging on page 1324
- voice (Application Profile) on page 1326

Copyright © 2019, Juniper Networks, Inc.

action

Syntax

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

Hierarchy Level

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

Release Information

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

Description

Discard traffic on a logical interface using tricolor marking policing.

NOTE: **This statement is supported only on IQ2 interfaces.**

The remaining statement is explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

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

Related Documentation

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

Copyright © 2019, Juniper Networks, Inc.

address (CoS on ATM Interfaces)

Syntax

```
address address {
  destination address;
}
```

Hierarchy Level

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

Release Information Statement introduced before Junos OS Release 7.4.

Description For CoS on ATM interfaces, configure the interface address.

Options ***address***—Address of the interface.

The remaining statements are explained separately.

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

Related Documentation

- Configuring CoS for ATM2 IQ Virtual Circuit Tunnels on page 850

Copyright © 2019, Juniper Networks, Inc.

adjust-minimum

Syntax

```
adjust-minimum rate;
```

Hierarchy Level

[edit class-of-service schedulers *scheduler-name*],
[edit class-of-service traffic-control-profiles *traffic-control-profile-name*]

Release Information

Statement introduced in Junos OS Release 11.4.

Description

For adjustments performed by the ANCP or multicast applications on EQ DPC, MIC, or MPC interfaces, specify the minimum shaping rate for an adjusted scheduler node. The node is associated with a traffic-control profile.

For adjustments performed by the multicast application on MIC or MPC interfaces, specify the minimum shaping rate for an adjusted queue. The queue is associated with a scheduler.

Options

rate—Minimum shaping rate for a node or a queue, in Mbps

Required Privilege Level

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

Related Documentation

- *Configuring the Minimum Adjusted Shaping Rate on Scheduler Nodes for Subscribers*

Copyright © 2019, Juniper Networks, Inc.

adjust-percent

Syntax adjust-percent *percentage*;

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

Release Information Statement introduced in Junos OS Release 11.4.

Description For a MIC or MPC interface, determine the percentage of adjustment for the shaping rate of a queue.

Options ***percentage***—Percentage of the shaping rate to adjust.

Range: 0 through 100 percent

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

Related Documentation

- *Configuring Shaping-Rate Adjustments on Queues*

Copyright © 2019, Juniper Networks, Inc.

application-profile

Syntax

```
application-profile profile-name {
  ftp {
    data {
      dscp (alias | bits);
      forwarding-class class-name;
    }
  }
  sip {
    video {
      dscp (alias | bits);
      forwarding-class class-name;
    }
    voice {
      dscp (alias | bits);
      forwarding-class class-name;
    }
  }
}
```

Hierarchy Level

```
[edit services cos],
[edit services cos rule rule-name term term-name then],
[edit services cos rule rule-name term term-name then reverse]
```

Release Information

Statement introduced in Junos OS Release 8.1.

Description

Define or apply a CoS application profile. When you apply a CoS application profile in a CoS rule, terminate the profile name with a semicolon (;).

Options

profile-name—Identifier for the application profile.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- *Configuring Application Profiles for Use as CoS Rule Actions*

Copyright © 2019, Juniper Networks, Inc.

application-sets (Services CoS)

Syntax applications-sets *set-name*;

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

Release Information Statement introduced in Junos OS Release 8.1.

Description Define one or more target application sets.

Options ***set-name***—Name of the target application set.

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

Related Documentation
- *Configuring Match Conditions In CoS Rules*

applications (Services CoS)

Syntax applications [*application-name*];

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

Release Information Statement introduced in Junos OS Release 8.1.

Description Define one or more applications to which the CoS services apply.

Options ***application-name***—Name of the target application.

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

Related Documentation
- Configuring CoS Rules on Services PICs on page 699
- *Configuring Match Conditions In CoS Rules*

Copyright © 2019, Juniper Networks, Inc.

atm-options

Syntax

```
atm-options {
  cell-bundle-size cells;
  ilmi;
  linear-red-profiles profile-name {
    high-plp-max-threshold percent;
    low-plp-max-threshold percent;
    queue-depth cells high-plp-threshold percent low-plp-threshold percent;
  }
  mpls {
    pop-all-labels {
      required-depth number;
    }
  }
  pic-type (atm1 | atm2);
  plp-to-clp;
  promiscuous-mode {
    vpi vpi-identifier;
  }
  scheduler-maps map-name {
    forwarding-class class-name {
      epd-threshold cells plp1 cells;
      linear-red-profile profile-name;
      priority (high | low);
      transmit-weight (cells number | percent number);
    }
    vc-cos-mode (alternate | strict);
  }
  use-null-cw;
  vpi vpi-identifier {
    maximum-vcs maximum-vcs;
    oam-liveness {
      up-count cells;
      down-count cells;
    }
    oam-period (disable | seconds);
    shaping {
      (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate burst
        length);
      queue-length number;
    }
  }
}
```

Hierarchy Level

```
[edit interfaces interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro Routers.

Description

Configure ATM-specific physical interface properties.

Copyright © 2019, Juniper Networks, Inc.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

NOTE: Certain options apply only to specific platforms.

Required Privilege Level

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

Related Documentation

- *Interface Encapsulations Overview*
- *multipoint-destination*
- shaping on page 1264
- vci on page 1321

Copyright © 2019, Juniper Networks, Inc.

atm-policer

Syntax

```
atm-policer policer-name {
  atm-service (cbr | rtvbr | nrtvbr | ubr) ;
  cdvt rate;
  logical-interface-policer;
  max-burst-size max-burst-size;
  peak-rate rate;
  policing-action (discard | discard-tag | count);
  sustained-rate rate;
}
```

Hierarchy Level

```
[edit firewall],
[edit interface interface-name unit unit-number]
```

Release Information

Statement introduced in Junos OS Release 12.1.

Description

(MX Series routers) Create a policer for each cell in the ATM packet. A policer defines the maximum traffic that can flow through an interface and further determines the actions to be taken when the traffic exceeds the defined limits.

When included at the **[edit firewall]** hierarchy level, the **atm-policer** statement creates a template, and you do not have to configure a policer individually for every ATM interface. To activate the ATM policer on an ATM interface, you must include the **atm-policer** statement in the **[edit interface]** hierarchy level.

Options

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

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- *Order of Policer and Firewall Filter Operations*
- *show class-of-service traffic-control-profile*
- show class-of-service interface on page 1364

Copyright © 2019, Juniper Networks, Inc.

atm-scheduler-map

Syntax atm-scheduler-map (*map-name* | default);

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Associate a scheduler map with a virtual circuit on a logical interface.

Options *map-name*—Name of scheduler map that you define at the **[edit interfaces *interface-name* atm-options scheduler-maps]** hierarchy level.

default—The default scheduler mapping.

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

Related Documentation

- *ATM2 IQ VC Tunnel CoS Components Overview*
- scheduler-maps (For ATM2 IQ Interfaces) on page 1260

 Copyright © 2019, Juniper Networks, Inc.

atm-service

Syntax

```
atm-service (cbr | rtvbr| nrtvbr);
```

Hierarchy Level

```
[edit class-of-service traffic-control-profiles traffic-control-profile-name]
[edit firewall atm-policer atm-policer-name]
```

Release Information Statement introduced in Junos OS Release 12.1.

Description (MX Series routers) Configure the ATM service category on ATM MICs to define bandwidth shaping and utilization. Shaping is based on the ATM service category.

Default If the ATM service category is not specified, bandwidth utilization is unlimited.

Options **cbr**—Use the constant bit rate.

nrtvbr—Use the non real-time variable bit rate.

rtvbr—Use the real-time variable bit rate.

NOTE: (MX Series with MPCs and ATM MICs with SFP) To configure up to OC12 CBR bandwidth speed per virtual circuit (VC) on an ATM MIC with SFP (MIC-3D-8OC3-2OC12-ATM), specify **cbr** as the ATM service category.

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

Related Documentation

- *show class-of-service traffic-control-profile*

Copyright © 2019, Juniper Networks, Inc.

buffer-size (Schedulers)

Syntax buffer-size (percent *percentage* | remainder | shared | temporal *microseconds*);

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

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.
shared option introduced in Junos OS Release 18.1 for PTX Series Packet Transport Routers.

Description Specify buffer size.

NOTE: On PTX Series Packet Transport Routers, buffer-size cannot be configured on rate-limited queues.

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, respectively.

Options **percent** ***percentage***—Buffer size as a percentage of the total buffer.

Range: 0 through 100

NOTE: For the routers with channelized OC12/STM4 IQE PIC with SFP (PB-4CHOC12-STM4-IQE-SFP) and channelized OC48/STM16 IQE PIC with SFP (PB-1CHOC48-STM16-IQE-SFP), the minimum buffer allocated to any queue is 18,432 bytes. If a queue is configured to have a buffer size less than 18K, the queue retains a buffer size of 18,432 bytes.

remainder—Remaining buffer available.

shared—On PTX Series routers, set a queue's buffer to be up to 100 percent of the interface's buffer. This option allows the queue's buffer to grow as large as 100 percent of the interface's buffer if and only if it is the only active queue for the interface.

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.

Copyright © 2019, Juniper Networks, Inc.

Range: The ranges vary by platform as follows:

- For SRX Series Services Gateways: 1 through 2,000,000 microseconds.
- For vSRX instances: 1 through 32,000,000 microseconds.

Required Privilege Level

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

Related Documentation

- Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size on page 365
- Table 47 on page 366

bypass-queuing-chip

Syntax bypass-queuing-chip;

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 14.2.
Statement introduced in Junos OS Release 18.2R1 on vMX routers.

Description On MPCs that contain a queuing chip and on vMX routers, bypass the queuing chip to increase the available bandwidth when rich queuing features are not required. When applied on a vMX router, this option saves a vCPU.

CAUTION: When enabling this option, do not oversubscribe the port bandwidth. Also, after committing this change on a vMX router, you must reboot the MPC that contains the interface.

NOTE: This option requires flexi-queuing-mode to be enabled.

NOTE: Once you apply this option, you can no longer apply schedulers or shapers to the interface.

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

Related Documentation

- Increasing Available Bandwidth on Rich-Queuing MPCs by Bypassing the Queuing Chip on page 940
- flexi-queuing-mode on page 1126

Copyright © 2019, Juniper Networks, Inc.

bytes (Dynamic Traffic Shaping)

Syntax

```
bytes bytes | $junos-cos-byte-adjust;
```

Hierarchy Level

```
[edit dynamic-profiles profile-name class-of-service traffic-control-profiles profile-name
  overhead-accounting],
[edit class-of-service traffic-control-profiles profile-name overhead-accounting]
```

Release Information Statement introduced in Junos OS Release 10.2.

Description Configure the number of overhead bytes.

Options ***bytes***—Byte adjustment value for the **cell-mode** or **frame-mode** shaping options. This can be the predefined variable **$junos-cos-byte-adjust**, which is the variable for byte adjustment 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.

BEST PRACTICE: We recommend using the cell-mode **cell-mode-bytes** *cell-mode-bytes* option or the frame-mode **frame-mode-bytes** *frame-mode-bytes* option rather than the **bytes** option.

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

Related Documentation

- *CoS Adjustment Control Profiles Overview*
- *Configuring Dynamic Shaping Parameters to Account for Overhead in Downstream Traffic Rates*
- Bandwidth Management for Downstream Traffic in Edge Networks Overview on page 984
- egress-shaping-overhead on page 1098

Copyright © 2019, Juniper Networks, Inc.

cbr

Syntax

```
cbr rate;
```

Hierarchy Level

```
[edit interfaces at-fpc/pic/port atm-options vpi vpi-identifier shaping],
[edit interfaces at-fpc/pic/port unit logical-unit-number address address family family
multipoint-destination address shaping],
[edit interfaces at-fpc/pic/port unit logical-unit-number shaping],
[edit logical-systems logical-system-name interfaces at-fpc/pic/port unit logical-unit-number
  address address family family multipoint-destination address shaping],
[edit logical-systems logical-system-name interfaces at-fpc/pic/port unit logical-unit-number
  shaping]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM encapsulation only, define a constant bit rate bandwidth utilization in the traffic-shaping profile.

Default Unspecified bit rate (UBR); that is, bandwidth utilization is unlimited.

Options ***rate***—Peak rate, in bits per second (bps) or cells per second (cps). 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). You can also specify a value in cells per second by entering a decimal number followed by the abbreviation **c**; values expressed in cells per second are converted to bits per second by means of the formula 1 cps = 384 bps.

For ATM1 and ATM2 OC3 interfaces, the maximum available rate is 100 percent of ***line-rate***, or 135,600,000 bps. For ATM1 OC12 interfaces, the maximum available rate is 50 percent of ***line-rate***, or 271,263,396 bps. For ATM2 IQ interfaces, the maximum available rate is 542,526,792 bps.

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

Related Documentation

- *Defining the ATM Traffic-Shaping Profile Overview*
- rtvbr on page 1253
- shaping on page 1264
- vbr on page 1319

Copyright © 2019, Juniper Networks, Inc.

cdvt

Syntax cdvt *rate*;

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

Release Information Statement introduced in Junos OS Release 12.1.

Description (MX Series routers) Define the Cell Delay/Variation Tolerance (CDVT) rate on a clear-channel multirate circuit emulation MIC.

Options *rate*—CDVT rate in microseconds.
Range: 1 through 1,800,000,000

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

Related Documentation
- *show class-of-service traffic-control-profile*

Copyright © 2019, Juniper Networks, Inc.

cell-mode (Dynamic Traffic Shaping)

Syntax

```
cell-mode (bytes bytes | $junos-cos-byte-adjust | cell-mode-bytes cell-mode-bytes
  |$junos-cos-byte-adjust-cell);
```

Hierarchy Level

```
[edit dynamic-profiles profile-name class-of-service traffic-control-profiles profile-name
  overhead-accounting],
[edit class-of-service traffic-control-profiles profile-name overhead-accounting],
```

Release Information

Statement introduced in Junos OS Release 10.2.
Variable ***$junos-cos-byte-adjust-cell*** introduced in Junos OS Release 13.1.

Description

Configure the mode to shape downstream ATM traffic as cells.

Options

bytes—Byte adjustment value for the **cell-mode** or **frame-mode** shaping options.

$junos-cos-byte-adjust—Predefined variable for byte adjustment 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.

cell-mode-bytes *cell-mode-bytes*—Shaping is based on the number of bytes in cells, and accounts for the ATM cell encapsulation and padding overhead. The resulting traffic stream conforms to the policing rates configured in downstream ATM switches, reducing the number of packet drops in the Ethernet network.

$junos-cos-byte-adjust-cell—Predefined variable for the cell mode shaping. This variable can not be used when the **overhead-accounting bytes *bytes*** option is configured.

BEST PRACTICE: We recommend using the **cell-mode-bytes *cell-mode-bytes*** option rather than the **bytes** option.

Range: –120 through 124 bytes

NOTE: If you specify a value for the **bytes *bytes*** option, you cannot specify a value for either the **cell-mode-bytes** option.

NOTE: Cell mode is supported only on logical interfaces and interface sets; it is not supported on physical interfaces (ifd or ifd-remaining).

Default: The default is frame-mode.

Copyright © 2019, Juniper Networks, Inc.

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

Related Documentation
- *CoS Adjustment Control Profiles Overview*
- *Configuring CoS Adjustment Control Profiles*
- *adjustment-control-profiles*
- *Configuring Dynamic Shaping Parameters to Account for Overhead in Downstream Traffic Rates*
- Bandwidth Management for Downstream Traffic in Edge Networks Overview on page 984
- egress-shaping-overhead on page 1098
- bytes on page 1059
- frame-mode on page 1144

class (CoS-Based Forwarding)

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

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

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Configure CoS-based forwarding class.

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

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation
- Overriding the Input Classification on page 250

Copyright © 2019, Juniper Networks, Inc.

class (Forwarding Classes)

Syntax

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

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 8.1.

Description

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

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

Options

class-name—Name of forwarding class.

queue-number—Output queue number.

Range: 0 through 7. Some T Series router PICs are restricted to 0 through 3.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- Configuring a Custom Forwarding Class for Each Queue on page 212
- queue (Global Queues) on page 1239

Copyright © 2019, Juniper Networks, Inc.

classification-override

Syntax

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

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Override the incoming packet classification, assigning all packets sent to a destination prefix to the same output transmission queue.

Required Privilege Level

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

Related Documentation

- Overriding the Input Classification on page 250
- **policy-statement** in the *Junos OS Routing Protocols Library*

Copyright © 2019, Juniper Networks, Inc.

classifiers (Definition)

Syntax

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

Hierarchy Level

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

Release Information

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

Description

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

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

Options

classifier-name—Name of the aggregate behavior classifier.

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

Required Privilege Level

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

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36

Copyright © 2019, Juniper Networks, Inc.

classifiers (Logical Interface)

Syntax

```
classifiers {
  type (classifier-name | default) family (mpls | inet);
  no-default;
}
```

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
no-default option added for MX Series devices only in Junos OS Release 16.1.

Description

Apply a CoS aggregate behavior classifier to a logical interface. You can apply a default classifier or one that is previously defined.

On MX Series devices, if you do not explicitly apply a classifier configuration to the interface, the default classifier is applied to the interface. Apply the **no-default** option to disable the application of any default classifier to the routing instance.

Options

classifier-name—Name of the aggregate behavior classifier.

type—Traffic type.

Values: **dscp**, **dscp-ipv6**, **exp**, **ieee-802.1**, **inet-precedence**

NOTE: You can only specify a family for the **dscp** and **dscp-ipv6** types.

Required Privilege Level

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

Related Documentation

- Default DSCP and DSCP IPv6 Classifiers on page 41
- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53

Copyright © 2019, Juniper Networks, Inc.

classifiers (Physical Interface)

Syntax

```
classifiers {
  type (classifier-name | default) ;
}
```

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.

Description

Apply a CoS aggregate behavior classifier to a physical interface. You can apply a default classifier or one that is previously defined.

Options

classifier-name—Name of the aggregate behavior classifier.

type—Traffic type.

Values: **dscp**, **ieee-802.1**, and **inet-precedence**

Required Privilege Level

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

Related Documentation

- dscp on page 1091
- inet-precedence on page 1162
- ieee-802.1 on page 1156

Copyright © 2019, Juniper Networks, Inc.

classifiers (Routing Instance)

Syntax

```
classifiers {
  exp (classifier-name | default);
  dscp (classifier-name | default);
  dscp-ipv6 (classifier-name | default);
  no-default;
}
```

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.
dscp and **dscp-ipv6** support introduced in Junos OS Release 9.6.
no-default option added for MX Series devices only in Junos OS Release 16.1.

Description

For routing instances with VRF table labels enabled, apply a custom MPLS EXP classifier or DSCP classifier to the routing instance. You can apply the default classifier or one that is previously defined.

If you do not explicitly apply a classifier configuration to the routing instance, the default classifier is applied to the routing instance. Apply the **no-default** option to disable the application of any default classifier to the routing instance.

Options

classifier-name—Name of the behavior aggregate MPLS EXP or DSCP classifier.

Required Privilege Level

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

Related Documentation

- Applying MPLS EXP Classifiers to Routing Instances on page 87
- Applying Behavior Aggregate Classifiers to Logical Interfaces on page 53

Copyright © 2019, Juniper Networks, Inc.

class-of-service

Syntax class-of-service { ... }

Hierarchy Level [edit]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure Junos CoS features.

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

- Understanding How Class of Service Manages Congestion and Controls Service Levels in the Network on page 4

Copyright © 2019, Juniper Networks, Inc.

class-of-service (Protocols MPLS)

Syntax

```
class-of-serviceclass-of-service cos-value;
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols mpls],
[edit logical-systems logical-system-name protocols mpls static-label-switched-path
  lsp-name ingress],
[edit logical-systems logical-system-name protocols mpls label-switched-path lsp-name],
[edit logical-systems logical-system-name protocols mpls label-switched-path lsp-name
  (primary | secondary) path-name],
[edit protocols mpls],
[edit protocols mpls label-switched-path lsp-name],
[edit protocols mpls label-switched-path lsp-name (primary | secondary) path-name],
[edit logical-systems logical-system-name protocols mpls static-label-switched-path
  lsp-name ingress]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 13.2X51-D15 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D30 for the QFX Series Virtual Chassis and Virtual Chassis Fabric.

Description

Class-of-service (CoS) value given to all packets in the LSP.

The CoS value might affect the scheduling or queuing algorithm of traffic traveling along an LSP.

Options

cos-value—CoS value. A higher value typically corresponds to a higher level of service.

Range: 0 through 7

Default: If you do not specify a CoS value, the IP precedence bits from the packet's IP header are used as the packet's CoS value.

Required Privilege Level

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

Related Documentation

- Configuring Class of Service for MPLS LSPs on page 79
- *Configuring the Ingress Router for Static LSPs*
- *Configuring the Intermediate (Transit) and Egress Routers for Static LSPs*

Copyright © 2019, Juniper Networks, Inc.

code-point-aliases

Syntax

```
code-point-aliases {
  type {
    alias-name bits;
  }
}
```

Hierarchy Level

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Define an alias for a CoS marker.

Options ***alias-name***—Name of the code-point alias.

bits—6-bit value of the code-point bits, in decimal form.

type—CoS marker type.
Values: **dscp**, **dscp-ipv6**, **exp**, **ieee-802.1**, **ieee-802.1ad**, **inet-precedence**

Usage Guidelines See "Defining Aliases for CoS Value Bit Patterns" on page 48.

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

Related Documentation

- Defining Aliases for CoS Value Bit Patterns on page 48

 Copyright © 2019, Juniper Networks, Inc.

code-point

Syntax code-point [*aliases*] [*bit-patterns*];

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

Release Information Statement introduced before Junos OS Release 7.4.

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

Options ***aliases***—Name of each alias.

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

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

Related Documentation

- Configuring Rewrite Rules on page 391

Copyright © 2019, Juniper Networks, Inc.

code-points (CoS)

Syntax

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

Hierarchy Level

```
[edit class-of-service classifiers type classifier-name forwarding-class class-name loss-priority
  level ]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.2 for SRX Series devices.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 12.1X44 for the SRX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.

Description

Specify one or more DSCP code-point aliases or bit sets to apply to a forwarding class..

NOTE: OCX Series switches do not support MPLS, and therefore, do not support EXP code points or code point aliases.

Options

aliases—Name of the DSCP alias.

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

Required Privilege Level

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

Related Documentation

- *Understanding Interfaces*
- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Example: Configuring Behavior Aggregate Classifiers on page 65
- *Example: Configuring BA Classifiers on Transparent Mode Security Devices*

Copyright © 2019, Juniper Networks, Inc.

copy-plp-all

Syntax
```
copy-plp-all;
```

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

Release Information Statement introduced in Junos OS Release 10.3.

Description Enable PLP bit copying for ingress and egress for unicast and multicast traffic when traffic is ingressing one FPC and egressing the other (from E3-FPC to non-E3 FPC on M320 routers, or from ES-FPC to non-ES FPC on T Series routers).

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

Related Documentation
- Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 361

copy-tos-to-outer

Syntax
```
copy-tos-to-outer service-type (gre | mt);
```

Hierarchy Level
```
[edit chassis]
```

Release Information Statement introduced in Junos OS Release 17.1.

Description For static and dynamic GRE tunnel interfaces and MT tunnel interfaces on MPCs, copy the inner IP header's ToS bits to the outer IP packet header for traffic transiting the router. This statement affects all MPC interfaces and takes precedence over the copy-tos-to-outer-ip-header-transit, which applies to individual interfaces.

Once committed, this configuration statement only affects new gr- and mt- interfaces. To affected an existing gr- or mt- interface, you must delete and re-add the interface.

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

Related Documentation
- copy-tos-to-outer-ip-header-transit on page 1077
- copy-tos-to-outer-ip-header on page 1076

Copyright © 2019, Juniper Networks, Inc.

copy-tos-to-outer-ip-header

Syntax

```
copy-tos-to-outer-ip-header;
```

Hierarchy Level

[edit interfaces gr-*fpc/pic/port* unit *logical-unit-number*],
[edit interfaces gre unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces gr-*fpc/pic/port* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces gre unit *logical-unit-number*]

Release Information

Statement introduced in Junos OS Release 8.2.
Support for GRE interfaces for Generalized MPLS (GMPLS) introduced in Junos OS Release 12.3R7.

Description

For GRE tunnel interfaces and GRE interfaces for GMPLS control channels only, enable the inner IP header's ToS bits to be copied to the outer IP packet header for traffic originating in the Routing Engine.

To verify that this option is enabled at the interface level, use the **show interfaces *interface-name* detail** command.

Default

If you omit this statement, the ToS bits in the outer IP header are set to 0.

Required Privilege Level

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

Related Documentation

- Configuring a GRE Tunnel to Copy ToS Bits to the Outer IP Header on page 695
- copy-tos-to-outer-ip-header-transit on page 1077

Copyright © 2019, Juniper Networks, Inc.

copy-tos-to-outer-ip-header-transit

Syntax

```
copy-tos-to-outer-ip-header-transit;
```

Hierarchy Level

```
[edit interfaces gr-fpc/pic/port unit logical-unit-number]
```

Release Information Statement introduced in Junos OS Release 17.1.

Description For static GRE tunnel interfaces on MPCs, copy the inner IP header's ToS bits to the outer IP packet header for traffic transiting the router.

Default If you omit this statement, the ToS bits in the outer IP header are set to 0.

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

Related Documentation

- copy-tos-to-outer-ip-header on page 1076

data (FTP)

Syntax

```
data {
  dscp (alias | bits);
  forwarding-class class-name;
}
```

Hierarchy Level

```
[edit services cos application-profile profile-name ftp]
```

Release Information Statement introduced in Junos OS Release 9.3.

Description Set the appropriate **dscp** and **forwarding-class** value for FTP data.

Default By default, the system will not alter the DSCP or forwarding class for FTP data traffic.

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

Related Documentation

- Configuring CoS Rules on Services PICs on page 699
- video (Application Profile) on page 1322
- voice (Application Profile) on page 1326

Copyright © 2019, Juniper Networks, Inc.

default (CoS Host Outbound Traffic)

Syntax default *value*;

Hierarchy Level [edit class-of-service host-outbound-traffic ieee-802.1]

Release Information Statement introduced in Junos OS Release 12.3.

Description Apply a global default value to the IEEE 802.1p—priority code point (PCP)—field in the Ethernet frame header for all host outbound traffic.

Options ***value***—Three-bit binary number.
Range: 000 through 111

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

Related Documentation

- Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic on page 597
- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388

Copyright © 2019, Juniper Networks, Inc.

delay-buffer-rate

Syntax delay-buffer-rate (percent *percentage* | *rate*);

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

Release Information Statement introduced in Junos OS Release 7.6.

Description For Gigabit Ethernet IQ, Channelized IQ PICs, and FRF.15 and FRF.16 LSQ interfaces only, 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. For more information, see Table 28 on page 277.

Options

percent *percentage*—For LSQ interfaces, delay-buffer rate as a percentage of the available interface bandwidth.

Range: 1 through 100 percent

rate—For IQ and IQ2 interfaces, 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 6,400,000,000,000 bps

NOTE: **Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.**

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

Related Documentation

- Oversubscribing Interface Bandwidth on page 273
- Providing a Guaranteed Minimum Rate on page 285
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- output-traffic-control-profile on page 1219

Copyright © 2019, Juniper Networks, Inc.

destination-address (Services CoS)

Syntax

```
destination-address (address | any-unicast) <except>;
```

Hierarchy Level

```
[edit services cos rule rule-name term term-name from]
```

Release Information

Statement introduced in Junos OS Release 8.1.
address option enhanced to support IPv4 and IPv6 addresses in Junos OS Release 8.5.

Description

Specify the destination address for rule matching.

Options

address—Destination IPv4 or IPv6 address or prefix value.

Required Privilege Level

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

Related Documentation

- Configuring CoS Rules on Services PICs on page 699
- *Configuring Match Conditions In CoS Rules*

Copyright © 2019, Juniper Networks, Inc.

destination (Interfaces)

Syntax

```
destination address;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number tunnel],
[edit interfaces interface-name unit logical-unit-number family inet address address],
[edit interfaces interface-name unit logical-unit-number tunnel],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number
  family inet address address]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description For CoS on ATM interfaces, specify the remote address of the connection.

For point-to-point interfaces only, specify the address of the interface at the remote end of the connection.

For tunnel and encryption interfaces, specify the remote address of the tunnel.

Options ***address***—Address of the remote side of the connection.

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

Related Documentation

- Configuring Linear RED Profiles on ATM Interfaces on page 859
- *Multilink and Link Services Logical Interface Configuration Overview*
- *Configuring Encryption Interfaces*
- *Configuring Traffic Sampling on MX, M and T Series Routers*
- *Configuring Flow Monitoring*
- *Configuring Unicast Tunnels*

Copyright © 2019, Juniper Networks, Inc.

discard (Forwarding Class)

Syntax

```
discard;
```

Hierarchy Level

```
[edit class-of-service forwarding-policy next-hop-map map-name forwarding-class
  class-name]
```

Release Information Statement introduced in Junos OS Release 9.1.

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

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

Related Documentation

- Configuring CoS-Based Forwarding on page 224
- non-lsp-next-hop on page 1216

Copyright © 2019, Juniper Networks, Inc.

drop-probability (Interpolated Value)

Syntax

```
drop-probability [values];
```

Hierarchy Level

```
[edit class-of-service drop-profiles profile-name interpolate]
```

Release Information

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

Description

Define values for interpolated drop probabilities. The maximum number of drop probability values supported per drop profile is based on the line card.

On EX Series switches, this statement is supported only on the EX9200 switch, EX8200 standalone switches, and EX8200 Virtual Chassis.

Options

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

Range: 0 through 100

Required Privilege Level

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

Related Documentation

- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351
- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355

Copyright © 2019, Juniper Networks, Inc.

drop-probability (Percentage)

Syntax drop-probability *percentage*;

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Define drop probability percentages. The maximum number of drop probability values supported per drop profile is based on the line card.

Options ***percentage***—Probability that a packet is dropped, expressed as a percentage. A value of 0 means that a packet is never dropped, and a value of 100 means that all packets are dropped.

Range: 0 through 100 percent

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

Related Documentation

- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351
- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355

Copyright © 2019, Juniper Networks, Inc.

drop-profile (Schedulers)

Syntax

```
drop-profile profile-name;
```

Hierarchy Level

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

Release Information

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

Description

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

Options

profile-name—Name of the drop profile.

Required Privilege Level

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

Related Documentation

- Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 359
- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351

Copyright © 2019, Juniper Networks, Inc.

drop-profile-map (Schedulers)

Syntax

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

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name]
```

Release Information

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

Description

Define the loss-priority value for a drop profile.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- Default Schedulers Overview on page 257
- Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 359

Copyright © 2019, Juniper Networks, Inc.

drop-profiles

Syntax

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

Hierarchy Level

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

Release Information

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

Description

Define drop profiles for RED.

For a packet to be dropped, it must match the drop profile. When a packet arrives, RED checks the queue fill level. If the fill level corresponds to a nonzero drop probability, the RED algorithm determines whether to drop the packet.

Options

profile-name—Name of the drop profile.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355

Copyright © 2019, Juniper Networks, Inc.

drop-timeout (Forwarding Class)

Syntax drop-timeout *milliseconds*;

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

Release Information Statement introduced in Junos OS Release 8.2.

Description Disable or set the resequencing timeout interval for each forwarding class of a multiclass MLPPP.

Default If you do not include this statement, the default sequencing timeouts for T1 speeds (500 ms) or lower (1500 ms) apply.

Options ***milliseconds***—Time to wait for fragments. A value of 0 disables the resequencing logic for that forwarding class.

Range: 0 through 500 milliseconds for bundles with bandwidths or T1 speeds or higher or 1500 ms for bundles with bandwidths of less than T1 speeds.

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

Related Documentation

- Configuring Drop Timeout Interval for Fragmentation by Forwarding Class on page 710

Copyright © 2019, Juniper Networks, Inc.

dscp (Services CoS)

Syntax

```
dscp (alias | bits);
```

Hierarchy Level

```
[edit services cos application-profile profile-name (ftp | sip) (data | video | voice)],
[edit services cos rule rule-name term term-name then],
[edit services cos rule rule-name term term-name then reverse]
```

Release Information Statement introduced in Junos OS Release 8.1.

Description Define the Differentiated Services code point (DSCP) mapping that is applied to the packets.

Options ***alias***—Name assigned to a set of CoS markers.

bits—Mapping value in the packet header.

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

Related Documentation

- *Configuring Actions in CoS Rules*.
- Configuring CoS Rules on Services PICs on page 699

Copyright © 2019, Juniper Networks, Inc.

dscp (CoS Classifiers)

Syntax

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

Hierarchy Level

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Define the diffserv code point (DSCP) mapping that is applied to the packets.

Options *classifier-name*—Name of the classifier.

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Applying DSCP Classifiers to MPLS Traffic on page 83

Copyright © 2019, Juniper Networks, Inc.

dscp (CoS Interfaces)

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

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

Release Information Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.

Description For ACX Series Universal Metro routers, map the DSCP field of the incoming packet to the forwarding class and packet loss priority based on the specified DSCP classifier.

Options ***classifier-name***—Name of the previously defined DSCP behavior aggregate classifier.

default—Default DSCP behavior aggregate classifier.

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

Copyright © 2019, Juniper Networks, Inc.

dscp (Multifield Classifier)

Syntax

```
dscp [0 | value];
```

Hierarchy Level

```
[edit firewall family family-name filter filter-name term term-name then]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

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

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

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

Options

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

Required Privilege Level

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

Related Documentation

- Configuring and Applying Tricolor Marking Policers on page 175

Copyright © 2019, Juniper Networks, Inc.

dscp (Rewrite Rules)

Syntax dscp (*rewrite-name* | default) protocol (inet-both | inet-outer | mpls);

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

Release Information Statement introduced before Junos OS Release 7.4.

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

Logical interfaces do not support multiple **dscp** rewrite rules for the same protocol.

DSCP and DSCP IPv6 rewrite rules are supported on M Series and T Series routers when non-queuing PICs are installed, but are disabled when queuing PICs are installed with the following exceptions:

- On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.
- On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.

DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.

DSCP rewrite rules are not supported on T Series routers when IQ, IQ2, IQE, IQ2E, SONET/SDH OC48/STM16 IQE, or PD-5-10XGE-SFPP PICs are installed.

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.

protocol inet-both—For gr- interfaces (GRE tunnels) on MPCs, rewrite the DSCP CoS value to both the inner and outer header for Unicast/Multicast IPv4 traffic. The first six bits of the CoS value are rewritten and the final two bits are taken from the incoming CoS value.

protocol inet-outer—For gr- interfaces on MPCs, rewrite the DSCP CoS value to the outer header for Unicast/Multicast IPv4 traffic. The first six bits of the CoS value are rewritten and the final two bits are taken from the incoming CoS value.

protocol mpls—(Optional for ingress MPLS tunnel nodes) For interfaces on MX Series routers or hosted on Enhanced III FPCs in M120 or M320 routers only, rewrite the MPLS EXP bits in the MPLS header independently of the IPv4 DSCP value for IPv4 packets entering an MPLS tunnel.

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

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring Rewrite Rules on page 391
- Applying Rewrite Rules to Output Logical Interfaces on page 401
- **protocol (Rewrite Rules) on page 1237**
- Rewriting MPLS and IPv4 Packet Headers on page 403
- **rewrite-rules (Definition) on page 1248**

dscp (Rewrite Rules on Physical Interface)

Syntax

```
dscp (rewrite-name | default);
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name rewrite-rules
```

Release Information Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.

Description Associate a rewrite-rules configuration or default mapping with a specific interface.

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

Copyright © 2019, Juniper Networks, Inc.

dscp-code-point (CoS Host Outbound Traffic)

Syntax dscp-code-point *value*;

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

Release Information Statement introduced in Junos OS Release 8.4.
Statement introduced before Junos OS Release 11.4 for EX Series switches.
Support for distributed protocol handler traffic introduced in Junos OS Release 13.2.

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

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

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

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

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

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

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Default DSCP and DSCP IPv6 Classifiers on page 41
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 241.

Copyright © 2019, Juniper Networks, Inc.

dscp-ipv6 (CoS Rewrite Rules)

Syntax

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

Hierarchy Level

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Support for **protocol mpls** option introduced in Junos OS Release 10.4R2.

Description

For IPv6 traffic, apply a DSCP rewrite rule.

Logical interfaces do not support multiple **dscp-ipv6** rewrite rules for the same protocol.

DSCP and DSCP IPv6 rewrite rules are supported on M Series and T Series routers when non-queuing PICs are installed, but are disabled when queuing PICs are installed with the following exceptions:

- On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.
- On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.

DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.

DSCP rewrite rules are not supported on T Series routers when IQ, IQ2, IQE, IQ2E, SONET/SDH OC48/STM16 IQE, or PD-5-10XGE-SFPP PICs are installed.

Options

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

default—Default mapping.

protocol mpls—(Optional for ingress MPLS tunnel nodes) For interfaces on MX Series routers or hosted on Enhanced III FPCs in M120 or M320 routers only, rewrite the MPLS EXP bits in the MPLS header independently of the IPv6 DSCP value for IPv6 packets entering an MPLS tunnel.

Required Privilege Level

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

Related Documentation

- Configuring Rewrite Rules on page 391
- protocol on page 1237
- Setting IPv6 DSCP and MPLS EXP Values Independently on page 398
- Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 398

Copyright © 2019, Juniper Networks, Inc.

- Applying Rewrite Rules to Output Logical Interfaces on page 401
- rewrite-rules (Definition) on page 1248

egress-policer-overhead

Syntax egress-policer-overhead *bytes*;

Hierarchy Level [edit chassis fpc *slot-number* pic *pic-number*]

Release Information Statement introduced before Junos OS Release 11.1.

Description Add the specified number of bytes to the actual length of an Ethernet frame when determining the actions of Layer 2 policers, MAC policers, or queue rate limits applied to output traffic on the line card. You can configure egress policer overhead to account for egress *shaping* overhead bytes added to output traffic on the line card.

On M Series and T Series routers, this statement is supported on Gigabit Ethernet Intelligent Queuing 2 (IQ2) PICs and Enhanced IQ2 (IQ2E) PICs. On MX Series routers, this statement is supported for interfaces configured on Dense Port Concentrators (DPCs).

NOTE: This statement is not supported on Modular Interface Cards (MICs) or Modular Port Concentrators (MPCs) in MX Series routers.

Options ***bytes***—Number of bytes added to a packet exiting an interface.
Range: 0–255 bytes
Default: 0

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

Related Documentation

- egress-shaping-overhead on page 1098
- Policer Overhead to Account for Rate Shaping Overview on page 195
- Example: Configuring Policer Overhead to Account for Rate Shaping on page 196
- Configuring a Policer Overhead on page 817
- CoS on Enhanced IQ2 PICs Overview on page 794

Copyright © 2019, Juniper Networks, Inc.

egress-shaping-overhead

Syntax egress-shaping-overhead *number*;

Hierarchy Level [edit chassis fpc *slot-number* pic *pic-number* traffic-manager],
[edit chassis lcc *number* fpc *slot-number* pic *pic-number* traffic-manager]

Release Information Statement introduced in Junos OS Release 8.3.

Description Number of bytes to add to packet to determine shaped session packet length.

NOTE: On M Series and T Series routers with Gigabit Ethernet Intelligent Queuing 2 (IQ2) PICs and Enhanced IQ2 (IQ2E) PICs and on MX Series routers with Dense Port Concentrators (DPCs) only, to account for egress shaping overhead bytes added to output traffic on the line card, you must use the **egress-policer-overhead** statement to explicitly configure corresponding egress policing overhead for Layer 2 policers, MAC policers, or queue rate limits applied to output traffic on the line card.

NOTE: For MIC and MPC interfaces on MX Series routers, by default the value of **egress-shaping-overhead** is configured to 20, which means that the number of class-of-service (CoS) shaping overhead bytes to be added to the packets is 20. The interfaces on DPCs in MX Series routers, the default value is zero. For interfaces on PICs other than the 10-port 10-Gigabit Oversubscribed Ethernet (OSE) Type 4, you should configure **egress-shaping-overhead** to a minimum of 20 bytes to add a shaping overhead of 20 bytes to the packets.

NOTE: When you change the **egress-shaping-overhead** value, on M Series, T Series, and MX104 routers the PIC on which it is changed is restarted. On MX5 routers, the MIC on which it is changed is restarted. On other MX Series routers, the DPC/MPC on which it is changed is restarted.

Options *number*—When traffic management (queuing and scheduling) is configured on the egress side, the number of CoS shaping overhead bytes to add to the packets on the egress interface.

Range:

- –63 through 192.

Copyright © 2019, Juniper Networks, Inc.

- –62 through 192 for vSRX.

NOTE: The L2 headers (DA/SA + VLAN tags) are automatically a part of the shaping calculation.

Required Privilege Level

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

Related Documentation

- egress-policer-overhead on page 1097
- Configuring CoS for L2TP Tunnels on ATM Interfaces on page 848
- ingress-shaping-overhead on page 1170
- mode (Layer 2 Tunneling Protocol Shaping) on page 1211, ingress-shaping-overhead on page 1170
- traffic-manager on page 1306

Copyright © 2019, Juniper Networks, Inc.

enhanced (forwarding-class-accounting)

Syntax

```
enhanced {
  overhead-bytes overhead-value  (0-255 bytes);
  traffic-type (ucast | mcast);
  family (ipv4 | ipv6 | both );
  direction (ingress | egress | both);
}
```

Hierarchy Level

```
[edit interfaces interface-name forwarding-class-accounting]
[edit interfaces interface-name  unit logical-unit-number forwarding-class-accounting]
```

Release Information Statement introduced in Junos OS Release 14.1 in MX Series.

Description Specify the traffic and type for which you want to apply counters. A single aggregate counter per forwarding class can be used for inet and inet6 flows. Forwarding class accounting applies to transit traffic only, not host-generated or host-bound traffic. For ingress, only packets forwarded to the fabric are counted. For egress, only packets forwarded to the WAN are counted. Non-relevant network protocols such as ARP, BFD, and EOAM, as well as dropped packets, are not counted.

NOTE: Forwarding class accounting (enhanced mode) is not supported in *hyper-mode (forwarding-options)*.

Options **overhead-bytes *overhead-value***—The number of overhead-bytes to be accounted.

Range: 0 through 255 bytes

Default: If zero, or no value is specified, the number of overhead-bytes will be determined by the encapsulation configured on the interface. For Ethernet, the default values are as follows:

Copyright © 2019, Juniper Networks, Inc.

Untagged DIX, includes CRC	18
Single-tagged DIX, includes CRC	22
Double-tagged DIX, includes CRC	26
VLAN-bridge (CRC)	4
VLAN-CCC (CRC)	4
Untagged VLAN-TCC, includes CRC	18
Single-tagged VLAN-TCC, includes CRC	22
Double-tagged VLAN-TCC, includes CRC	26
VLAN-VPLS (CRC)	4

Packet over SONET (POS) interfaces on an MPC are dependent on the frame check sequence (FCS) settings on the interface, which may be two or four bytes. The default values for POS interfaces are as follows:

PPP (Layer 2 overhead plus FCS)	4+ (2 or 4)
Cisco-HDLC (Layer 2 overhead plus FCS)	4+ (2 or 4)
Frame-relay (Layer 2 overhead plus FCS)	4+ (2 or 4)
Cisco-HDLC-CCC (FCS)	2 or 4
Cisco-HDLC-TCC (FCS)	2 or 4
PPP-CCC (FCS)	2 or 4
PPP-TCC (FCS)	2 or 4
Frame-relay-CCC (FCS)	2 or 4
Frame-relay-TCC (FCS)	2 or 4

traffic-type *(ucast | mcast)*—Traffic to be counted can be unicast only or multicast only.

Default: not enabled, i.e., both unicast and multicast are counted

family *(ipv4 | ipv6 | both)*—Traffic to be counted can be IPv4, IPv6, or both.

Default: both

direction *(ingress | egress | both)*—Traffic can be inbound to the interface, outbound, or both.

Default: both

Copyright © 2019, Juniper Networks, Inc.

Required Privilege Level

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

Related Documentation

- CoS-Based Interface Counters for IPv4 or IPv6 Aggregate on Layer 2 on page 593
- show class-of-service interface on page 1364
- *clear interfaces statistics*

Copyright © 2019, Juniper Networks, Inc.

enhanced-priority-mode

Syntax

```
enhanced-priority-mode;
```

Hierarchy Level

[edit chassis fpc *slot-number* traffic-manager]

Release Information

Statement introduced in Junos OS Release 16.1 for MPC7E-MRATE, MPC7E-10G, MPC8E, and MPC9E.
Statement introduced in Junos OS Release 18.1 for PTX Series.

Description

Enable the enhanced priority mode. When you enable the enhanced priority mode, the scheduler supports four additional per-priority shaping rates and two additional excess priorities at the interface and interface set level. The four additional per-priority shaping rates are: Guaranteed Strict-high, Guaranteed Medium-low, Excess medium-high, and Excess medium-low. The two additional excess priorities are: Excess-rate Medium-high and Excess-rate Medium-low.

NOTE: **The line card reboots when you enable or delete the enhanced priority mode feature.**

Required Privilege Level

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

Related Documentation

- excess-rate-medium-high on page 1108
- excess-rate-medium-low on page 1109
- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- shaping-rate-excess-medium-high on page 1275
- shaping-rate-excess-medium-low on page 1276
- shaping-rate-priority-medium-low on page 1282
- shaping-rate-priority-strict-high on page 1283
- traffic-manager on page 1306
- traffic-control-profiles on page 1305

Copyright © 2019, Juniper Networks, Inc.

epd-threshold

Syntax

```
epd-threshold cells plp1 cells;
```

Hierarchy Level

```
[edit interfaces at-fpc/pic/port atm-options scheduler-maps map-name forwarding-class
  class-name]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM2 IQ interfaces only, define the EPD threshold on a VC. The EPD threshold is a limit on the number of transmit packets that can be queued. Packets that exceed the limit are discarded.

Default If you do not include either the **epd-threshold** or the **linear-red-profile** statement in the forwarding class configuration, the Junos OS uses an EPD threshold based on the available bandwidth and other parameters.

Options ***cells***—Maximum number of cells.

Range: For 1-port and 2-port OC12 interfaces, 1 through 425,984 cells. For 1-port OC48 interfaces, 1 through 425,984 cells.For 2-port OC3, DS3, and E3 interfaces, 1 through 212,992 cells. For 4-port DS3 and E3 interfaces, 1 through 106,496 cells.

plp1 ***cells***—Early packet drop threshold value for PLP 1.

Range: For 1-port and 2-port OC12 interfaces, 1 through 425,984 cells. For 1-port OC48 interfaces, 1 through 425,984 cells.For 2-port OC3, DS3, and E3 interfaces, 1 through 212,992 cells. For 4-port DS3 and E3 interfaces, 1 through 106,496 cells.

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

Related Documentation

- Applying Scheduler Maps to ATM Interfaces on page 853
- linear-red-profile on page 1188

Copyright © 2019, Juniper Networks, Inc.

excess-bandwidth-share

Syntax excess-bandwidth-share (proportional *value* | equal);

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

Release Information Statement introduced in Junos OS Release 8.5.

Description Specify the method of sharing excess bandwidth in a hierarchical scheduler environment. With hierarchical schedulers, you can provide shaping and scheduling at the service VLAN level as well as other levels, such as the physical interface. You can also group a set of logical interfaces and then apply scheduling and shaping parameters to the logical interface set.

To configure CoS hierarchical schedulers, you must enable hierarchical scheduling by including the hierarchical-scheduler statement at the **[edit interfaces]** hierarchy for the physical interface. If you do not include this statement, the interfaces on the MX Series router cannot use hierarchical interfaces.

The Enhanced Queuing DPC supports the following hierarchical scheduler characteristics:

- Shaping at the physical interface level
- Shaping and scheduling at the service VLAN interface set level
- Shaping and scheduling at the customer VLAN logical interface level
- Scheduling at the queue level

Options **equal**—Share excess bandwidth equally among the configured interfaces.

proportional *value*—(Default) Share excess bandwidth proportionally according to the specified value. In this mode, the excess bandwidth is shared at the ratio of the logical interface shaping rates.

Default: 32.64 Mbps

This example sets the excess bandwidth sharing for an Enhanced Queuing DPC interface proportionally at a rate of 100 Mbps and a shaping rate of 80 Mbps applied to the interface through the output-traffic-control profile for scheduling and shaping:

```
[edit class-of-service interfaces interface-set example-interface-set]
user@host# set excess-bandwidth-share proportional 100m
user@host# set output-traffic-control-profile PIR-80Mbps
```

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

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring Hierarchical Schedulers for CoS on page 342
- Configuring Interface Sets on page 264
- Enhanced Queuing DPC CoS Properties on page 908
- Configuring MDRR on Enhanced Queuing DPCs on page 913

excess-priority

Syntax

```
excess-priority [ low | medium-low | medium-high | high | none];
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name]
```

Release Information

Statement introduced in Junos OS Release 9.3.
Option **none** introduced in Junos OS Release 11.4.

Description

Determine the priority of excess bandwidth traffic on a scheduler.

NOTE: For Link Services IQ (LSQ) PICs or Multiservices PIC (MS-PICs), the **excess-priority** statement is allowed for consistency, but ignored. If an explicit priority is not configured for these interfaces, a default low priority is used. This default priority is also used in the excess region.

Options

low—Excess traffic for this scheduler has low priority.

medium-low—Excess traffic for this scheduler has medium-low priority.

medium-high—Excess traffic for this scheduler has medium-high priority.

high—Excess traffic for this scheduler has high priority.

none—System does not demote the priority of guaranteed traffic when the bandwidth exceeds the shaping rate or the guaranteed rate.

Required Privilege Level

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

Related Documentation

- Configuring Excess Bandwidth Sharing on IQE PICs on page 753
- Bandwidth Sharing on Nonqueuing Packet Forwarding Engines Overview on page 298
- Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 986

Copyright © 2019, Juniper Networks, Inc.

excess-rate

Syntax

```
excess-rate (percent percentage | proportion value);
```

Hierarchy Level

[edit class-of-service schedulers *scheduler-name*],
[edit class-of-service traffic-control-profiles *traffic-control-profile-name*]

Release Information

Statement introduced in Junos OS Release 9.3.
Application to the Multiservices PIC added in Junos OS Release 9.5.
Application to the MIC and MPC interfaces added in Junos OS Release 10.1.
Statement introduced in Junos OS Release 12.1X48R2 for PTX Series Packet Transport Routers.

Description

For an Enhanced IQ PIC interfaces, Multiservices PIC interfaces, or MX Series router interfaces on MPCs or MICs, and T4000 router interfaces on Type 5 FPCs and EX Series switches, determine the percentage or proportion of excess bandwidth traffic to share.

NOTE: The proportion option provides a greater range of values over the percent option and hence influences the priorities assigned to the queues.

Options

percentage—Percentage of the excess bandwidth to share.

Range: 0 through 100 percent

Default: Excess bandwidth is shared in proportion to the configured transmit rate of each queue.

value—(M Series, MX Series, T Series routers and EX Series switches only) Proportion of the excess bandwidth to share.

Range: 0 through 1000

NOTE: The proportion of excess bandwidth on MPC2-3D MPCs can be configured with increments of 1 from 0 through 1000. All other MPCs should be configured with increments of 10 from 0 through 1000.

Required Privilege Level

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

Related Documentation

- Configuring Scheduler Transmission Rate on page 282
- Configuring Excess Bandwidth Sharing on IQE PICs on page 753

Copyright © 2019, Juniper Networks, Inc.

- Allocating Excess Bandwidth Among Frame Relay DLCIs on Multiservices PICs on page 716
- Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 986

excess-rate-medium-high

Syntax excess-rate-medium-high (percent *percentage* | proportion *value*);

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

Release Information Statement introduced in Junos OS Release 16.1.

Description For MPC7E-MRATE, MPC7E-10G, MPC8E, and MPC9E interfaces, determine the percentage of excess bandwidth from high-priority traffic to share.

Options *percentage*—Percentage of the excess bandwidth to share.
Range: 1 through 100 percent

proportion—Proportion of the excess bandwidth to share.
Range: 0 through 1000

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

Related Documentation

- enhanced-priority-mode on page 1103
- excess-rate-medium-low on page 1109
- Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 986
- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- shaping-rate-excess-medium-high on page 1275
- shaping-rate-excess-medium-low on page 1276
- shaping-rate-priority-medium-low on page 1282
- shaping-rate-priority-strict-high on page 1283

Copyright © 2019, Juniper Networks, Inc.

excess-rate-medium-low

Syntax excess-rate-medium-low (percent *percentage* | proportion *value*);

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

Release Information Statement introduced in Junos OS Release 16.1.

Description For MPC7E-MRATE, MPC7E-10G, MPC8E, and MPC9E interfaces, determine the percentage of excess bandwidth from low-priority traffic to share.

Options ***percentage***—Percentage of the excess bandwidth to share.
Range: 1 through 100 percent

value—Proportion of the excess bandwidth to share.
Range: 0 through 1000

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

Related Documentation

- enhanced-priority-mode on page 1103
- excess-rate-medium-high on page 1108
- Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 986
- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- shaping-rate-excess-medium-high on page 1275
- shaping-rate-excess-medium-low on page 1276
- shaping-rate-priority-medium-low on page 1282
- shaping-rate-priority-strict-high on page 1283

Copyright © 2019, Juniper Networks, Inc.

excess-rate-high

Syntax excess-rate-high (percent *percentage* | proportion *value*);

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

Release Information Statement introduced in Junos OS Release 11.4.

Description For a MIC or MPC interface, determine the percentage of excess bandwidth from high-priority traffic to share.

Options ***percentage***—Percentage of the excess bandwidth to share.

Range: 0 through 100 percent

proportion—Proportion of the excess bandwidth to share.

Range: 0 through 1000

NOTE: The proportion of excess bandwidth on MPC2-3D MPCs can be configured with increments of 1 from 0 through 1000. All other MPCs should be configured with increments of 10 from 0 through 1000.

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

Related Documentation

- Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 986

 Copyright © 2019, Juniper Networks, Inc.

excess-rate-low

Syntax excess-rate-low (percent *percentage* | proportion *value*);

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

Release Information Statement introduced in Junos OS Release 11.4.

Description For a MIC or MPC interface, determine the percentage of excess bandwidth from low-priority traffic to share.

Options ***percentage***—Percentage of the excess bandwidth to share.

Range: 0 through 100 percent

value—Proportion of the excess bandwidth to share.

Range: 0 through 1000

NOTE: The proportion of excess bandwidth on MPC2-3D MPCs can be configured with increments of 1 from 0 through 1000. All other MPCs should be configured with increments of 10 from 0 through 1000.

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

Related Documentation

- Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 986

Copyright © 2019, Juniper Networks, Inc.

exclude-queue-overhead-bytes

Syntax

```
exclude-queue-overhead-bytes {
  include-hierarchy;
}
```

Hierarchy Level

[edit class-of-service interfaces *interface-name*],

Release Information

Statement introduced in Junos OS Release 18.2R1.

Description

By default, the Layer 2 header bytes applied to upper-level logical interfaces are included in CoS per-queue statistics at the physical interface, which can provide inaccurate results. This options enables you to exclude the counting of overhead bytes from aggregate queue statistics.

Options

include-hierarchy—Exclude the counting of overhead bytes from aggregate queue statistics of all child interfaces, including logical interfaces and interface sets, as well.

Required Privilege Level

interface

Related Documentation

- show class-of-service interface on page 1364

Copyright © 2019, Juniper Networks, Inc.

exp

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

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

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

Description Apply an MPLS experimental (EXP) rewrite rule.

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

default—The default mapping.

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

On interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series 5G Universal Routing Platforms and EX Series switches, we highly recommend that you configure the **default** option when you configure a behavior aggregate (BA) classifier that does not include a specific rewrite rule for MPLS packets. Doing so ensures that MPLS exp value is rewritten according to the BA classifier rules configured for forwarding or packet loss priority. This does not apply to ACX Series Universal Metro routers.

protocol-types—Specify one or more protocol matching criteria:

- **mpls-any**—Apply to MPLS packets, write MPLS header only.
- **mpls-inet-both**—Apply to IPv4 MPLS packets, write MPLS and IPv4 header.
- **mpls-inet-both-non-vpn**—Apply to IPv4 MPLS packets, write MPLS and IPv4 header for only non VPN traffic.

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

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring Rewrite Rules on page 391
- Rewriting the EXP Bits of All Three Labels of an Outgoing Packet on page 407
- Applying Rewrite Rules to Output Logical Interfaces on page 401
- protocol (Rewrite Rules) on page 1237
- rewrite-rules (Definition) on page 1248

exp-push-push-push

Syntax

```
exp-push-push-push default;
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description For M Series routers and EX Series switches, rewrite the EXP bits of all three labels of an outgoing packet, thereby maintaining CoS of an incoming non-MPLS packet.

Options **default**—Apply the default MPLS EXP rewrite table.

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

Related Documentation

- Rewriting the EXP Bits of All Three Labels of an Outgoing Packet on page 407
- dscp (Rewrite Rules) on page 1093
- dscp-ipv6 (CoS Rewrite Rules) on page 1096
- exp on page 1113
- exp-swap-push-push on page 1115
- ieee-802.1 (Rewrite Rules on Logical Interface) on page 1158
- ieee-802.1ad on page 1160
- inet-precedence (CoS Rewrite Rules) on page 1164
- rewrite-rules (Definition) on page 1248

Copyright © 2019, Juniper Networks, Inc.

exp-swap-push-push

Syntax

```
exp-swap-push-push default;
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For M Series routers and EX Series switches, rewrite the EXP bits of all three labels of an outgoing packet, thereby maintaining CoS of an incoming MPLS packet.

Options

default—Apply the default MPLS EXP rewrite table.

Required Privilege Level

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

Related Documentation

- Rewriting the EXP Bits of All Three Labels of an Outgoing Packet on page 407
- dscp (Rewrite Rules) on page 1093
- dscp-ipv6 (CoS Rewrite Rules) on page 1096
- exp on page 1113
- exp-push-push-push on page 1114
- ieee-802.1 (Rewrite Rules on Logical Interface) on page 1158
- ieee-802.1ad on page 1160
- inet-precedence (CoS Rewrite Rules) on page 1164
- rewrite-rules (Definition) on page 1248

Copyright © 2019, Juniper Networks, Inc.

explicit-null-cos

Syntax

```
explicit-null-cos {
  inet;
  inet6;
}
```

Hierarchy Level

```
[edit forwarding-options]
```

Release Information

Statement introduced in Junos OS Release 18.1 on PTX Series routers with third-generation FPCs (FPC3).

Description

The default classification for explicit-null packets is based on the payload (IPv4 or IPv6 DSCP bits). Configure this to make the packet classification based on the MPLS EXP value rather than on the payload, thus preserving the MPLS classification of the packet.

Options

NOTE: Include both options below or neither to use the MPLS EXP value for classification regardless of the payload type.

inet—Use the MPLS EXP value for classification when the payload is an IPv4 packet.

inet6—Use the MPLS EXP value for classification when the payload is an IPv6 packet.

Required Privilege Level

interface

Related Documentation

- Applying MPLS EXP Classifiers for Explicit-Null Labels on page 94

Copyright © 2019, Juniper Networks, Inc.

fabric (Class-of-Service)

Syntax

```
fabric {
  scheduler-map {
    priority (high | low) scheduler scheduler-name;
  }
}
```

Hierarchy Level

[edit class-of-service]

Release Information

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

Description

Define CoS parameters of the switch fabric. For M320 routers, MX Series routers, T Series routers and EX Series switches only, associate a scheduler with a fabric priority.

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

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- Associating Schedulers with Fabric Priorities on page 331

Copyright © 2019, Juniper Networks, Inc.

family (CoS on ATM Interfaces)

Syntax

```
family family {
  address address {
    destination address;
  }
}
```

Hierarchy Level

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

Release Information Statement introduced before Junos OS Release 7.4.

Description For CoS on ATM interfaces, configure the protocol family.

Options *family*—Protocol family:

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

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- CoS on ATM Interfaces Overview on page 841

Copyright © 2019, Juniper Networks, Inc.

family (Multifield Classifier)

Syntax

```
family family-name {
  filter filter-name {
    term term-name {
      ... term_configuration ...
    }
  }
}
```

Hierarchy Level

[edit firewall]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

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

Options

family-name—Protocol family:

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

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- Configuring Multifield Classifiers on page 98

Copyright © 2019, Juniper Networks, Inc.

fill-level (Drop Profiles)

Syntax fill-level *percentage*;

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

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

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

Options *percentage*—How full the queue is, expressed as a percentage. You configure the **fill-level** and **drop-probability** statements in pairs. To specify multiple fill levels, include multiple **fill-level** and **drop-probability** statements. The values you assign to each statement pair must increase relative to the previous pair's values. This is shown in the discrete graph in "Managing Congestion Using RED Drop Profiles and Packet Loss Priorities" on page 351.

Range: 0 through 100 percent

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

Related Documentation

- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351
- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355

 Copyright © 2019, Juniper Networks, Inc.

fill-level (Interpolated Value)

Syntax

```
fill-level [values];
```

Hierarchy Level

```
[edit class-of-service drop-profiles profile-name interpolate]
```

Release Information

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

Description

Define up to 64 values for interpolating queue fill level.

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

Options

values—Data points for mapping queue fill percentage.

Range: 0 through 100

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

Required Privilege Level

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

Related Documentation

- Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 351
- Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355.

Copyright © 2019, Juniper Networks, Inc.

filter (Applying to an Interface)

Syntax

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

Hierarchy Level

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

Release Information Statement introduced before Junos OS Release 7.4.

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

Options

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

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

Required Privilege Level

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

Related Documentation

- simple-filter on page 1287
- Configuring and Applying Tricolor Marking Policers on page 175
- Example: Classifying Packets Based on Their Destination Address on page 109
- Example: Configuring and Verifying a Complex Multifield Filter on page 111
- Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets on page 238
- Configuring a Simple Filter on page 839
- Configuring Policers Based on Logical Interface Bandwidth on page 122
- Effect of Two-Color Policers on Shaping Rate Changes on page 120

Copyright © 2019, Juniper Networks, Inc.

filter (Applying to a Logical Interface)

Syntax

```
filter {
  group filter-group-number;
  input filter-name;
  input-list [ filter-names ];
  output filter-name;
  output-list [ filter-names ];
}
```

Hierarchy Level

Protocol-independent firewall filter on MX Series router logical interface:

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

All other standard firewall filters on all other devices:

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

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Apply a stateless firewall filter to a logical interface at a specific protocol level.

Options

group *filter-group-number*—(Only Ex, M, MX, and T Series) Number of the group to which the interface belongs. Range: 1 through 255

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

input-list [*filter-names*]—Names of filters to evaluate when packets are received on the interface. Up to 16 filters can be included in a filter input list.

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

output-list [*filter-names*]—Names of filters to evaluate when packets are transmitted on the interface. Up to 16 filters can be included in a filter output list.

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 Firewall Filters*
- *Guidelines for Applying Standard Firewall Filters*

Copyright © 2019, Juniper Networks, Inc.

filter (Configuring)

Syntax

```
filter filter-name {
  accounting-profile name;
  enhanced-mode;
  fast-lookup-filter;
  filter-list-template;
  interface-shared;
  interface-specific;
  physical-interface-filter;
  promote gre-key;
  term term-name {
    ... term configuration ...
  }
}
```

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.
Logical systems support introduced in Junos OS Release 9.3.
physical-interface-filter statement introduced in Junos OS Release 9.6.
Support for the **interface-shared** statement introduced in Junos OS Release 12.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Configure firewall filters.

Options

filter-name—Name that identifies the filter. This must be a non-reserved string of not more than 64 characters. To include spaces in the name, enclose it in quotation marks (" "). Firewall filter names are restricted from having the form __.*__ (beginning and ending with underscores) or __.* (beginning with an underscore.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

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

Related Documentation

- *Guidelines for Configuring Firewall Filters*
- *Guidelines for Applying Standard Firewall Filters*
- Configuring Multifield Classifiers on page 98
- Using Multifield Classifiers to Set Packet Loss Priority on page 101
- simple-filter on page 1288

Copyright © 2019, Juniper Networks, Inc.

firewall

Syntax

```
firewall {
  atm-policer atm-policer-name {
    ... atm-policer-configuration ...
  }
  family protocol-family-name {
    ... protocol-family-configuration ...
  }
  filter ipv4-filter-name {
    ... ipv4-filter-configuration ...
  }
  hierarchical-policer hierarchical-policer-name {
    ... hierarchical-policer-configuration ...
  }
  interface-set interface-set-name {
    ... interface-set-configuration ...
  }
  policer two-color-policer-name {
    ... two-color-policer-configuration ...
  }
  three-color-policer three-color-policer-name {
    ... three-color-policer-configuration ...
  }
}
```

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.
Logical systems support introduced in Junos OS Release 9.3.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Configure firewall filters.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- *Guidelines for Configuring Firewall Filters*
- *Guidelines for Configuring Service Filters*
- *Guidelines for Configuring Simple Filters*
- Configuring Multifield Classifiers on page 98

Copyright © 2019, Juniper Networks, Inc.

- Using Multifield Classifiers to Set Packet Loss Priority on page 101

flexible-queuing-mode

Syntax **flexible-queuing-mode;**

Hierarchy Level [edit chassis]

Release Information Statement introduced in Junos OS Release 14.1R1 for MX Series Routers.

Description Enable flexible queuing on a non-HQoS MPCE that is installed in an MPC slot. A maximum of up to 32,000 queues are supported per port and per card, including queues on both ingress and egress interfaces.

When flexible queuing is enabled, the non-HQoS MPC is restarted for the changes to take effect and is brought online only if the power required for the queuing component is available in the power entry module (PEM). The MPC remains offline if the PEM cannot meet the power requirement for the queuing component.

You can configure flexible queuing even when a non-HQoS MPC is not present in the chassis. The configuration takes effect when a non-HQoS MPC is installed.

For more information about the MPCs and the Junos OS release that support this feature, see "Flexible Queuing Mode" on page 942.

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

Related Documentation

- Upgrading non-HQoS MPCs to Support Flexible Queuing on page 943
- Flexible Queuing Mode Overview on page 942
- *MPC3E on MX Series Routers Overview*
- *MPC5E on MX Series Routers Overview*
- *Protocols and Applications Supported by the MPC5E for MX Series Routers*
- Upgrading non-HQoS MPCs to Support Flexible Queuing on page 943

Copyright © 2019, Juniper Networks, Inc.

flexible-vlan-tagging

Syntax

```
flexible-vlan-tagging;
```

Hierarchy Level

```
[edit interfaces aex],
[edit interfaces ge-fpc/pic/port],
[edit interfaces et-fpc/pic/port],
[edit interfaces ps0],
[edit interfaces xe-fpc/pic/port]
```

Release Information

Statement introduced in Junos OS Release 8.1.
Support for aggregated Ethernet added in Junos OS Release 9.0.
Statement introduced in Junos OS Release 12.1x48 for PTX Series Packet Transport Routers.
Statement introduced in Junos OS Release 13.2X50-D15 for EX Series switches.
Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.

Description

Support simultaneous transmission of 802.1Q VLAN single-tag and dual-tag frames on logical interfaces on the same Ethernet port, and on pseudowire logical interfaces.

This statement is supported on M Series and T Series routers, for Fast Ethernet and Gigabit Ethernet interfaces only on Gigabit Ethernet IQ2 and IQ2-E, IQ, and IQE PICs, and for aggregated Ethernet interfaces with member links in IQ2, IQ2-E, and IQ PICs or in MX Series DPCs, or on Ethernet interfaces for PTX Series Packet Transport Routers or 100-Gigabit Ethernet Type 5 PIC with CFP.

This statement is supported on Gigabit Ethernet, 10-Gigabit Ethernet, 40-Gigabit Ethernet, and aggregated Ethernet interfaces on EX Series and QFX Series switches.

Required Privilege Level

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

Related Documentation

- *Enabling VLAN Tagging*
- *Configuring Flexible VLAN Tagging on PTX Series Packet Transport Routers*
- *Configuring Double-Tagged VLANs on Layer 3 Logical Interfaces*

Copyright © 2019, Juniper Networks, Inc.

forwarding-class (Services PIC Classifiers)

Syntax

```
forwarding-class class-name;
```

Hierarchy Level

```
[edit services cos application-profile profile-name (ftp | sip) (data | video | voice)],
[edit services cos rule rule-name term term-name then],
[edit services cos rule rule-name term term-name then reflexive; | revert; | reverse {]
```

Release Information Statement introduced in Junos OS Release 8.1.

Description Define the forwarding class to which packets are assigned.

Options ***class-name***—Name of the target application.

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

Related Documentation

- Configuring CoS Rules on Services PICs on page 699

Copyright © 2019, Juniper Networks, Inc.

forwarding-class (ATM2 IQ Scheduler Maps)

Syntax

```
forwarding-class class-name {
  epd-threshold cells plp1 cells;
  linear-red-profile profile-name;
  priority (high | low);
  transmit-weight (cells number | percent number);
}
```

Hierarchy Level

```
[edit interfaces at-fpc/pic/port atm-options scheduler-maps map-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For ATM2 IQ interfaces only, define forwarding class name and option values.

Options

class-name—Name of forwarding class.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- *ATM2 IQ VC Tunnel CoS Components Overview*
- Applying Scheduler Maps to ATM Interfaces on page 853

Copyright © 2019, Juniper Networks, Inc.

forwarding-class (BA Classifiers)

Syntax

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

Hierarchy Level

```
[edit class-of-service classifiers type classifier-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.

Description

Define forwarding class name and option values.

Options

class-name—Name of the forwarding class.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- Configuring Behavior Aggregate Classifiers on page 51

Copyright © 2019, Juniper Networks, Inc.

forwarding-class (CoS Host Outbound Traffic)

Syntax

```
forwarding-class class-name;
```

Hierarchy Level

```
[edit class-of-service host-outbound-traffic]
```

Release Information

Statement introduced in Junos OS Release 8.4.
Statement introduced before Junos OS Release 11.4 for EX Series switches.
Support for distributed protocol handler traffic introduced in Junos OS Release 13.2.

Description

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

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

This statement does not affect transit traffic or incoming traffic.

Default

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

Options

class-name—Name of the forwarding class.

Required Privilege Level

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

Related Documentation

- Understanding How Forwarding Classes Assign Classes to Output Queues on page 205
- Default Routing Engine Protocol Queue Assignments on page 234
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 241.

Copyright © 2019, Juniper Networks, Inc.

forwarding-class (Forwarding Policy)

Syntax

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

Hierarchy Level

```
[edit class-of-service forwarding-policy next-hop-map map-name]
[edit class-of-service forwarding-policy class class-name classification-override]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced for QFX10000 Series switches in Junos OS Release 17.1R1.
non-labelled-next-hop option introduced in Junos OS Release 19.1R1 for all platforms.

Description

Define forwarding class name and associated next hops.

Options

class-name—Name of the forwarding class.

non-labelled-next-hop—Match any non-labelled next hop.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- Overriding the Input Classification on page 250
- forwarding-class-default (Forwarding Policy) on page 1137

Copyright © 2019, Juniper Networks, Inc.

forwarding-class (Fragmentation)

Syntax

```
forwarding-class class-name {
  drop-timeout milliseconds;
  fragment-threshold bytes;
  multilink-class number;
  no-fragmentation;
}
```

Hierarchy Level

```
[edit class-of-service fragmentation-maps map-name];
```

Release Information Statement introduced before Junos OS Release 7.4.

Description For Multiservices and Services PICs link services IQ interfaces (**lsq**) only, define a forwarding class name and associated fragmentation properties within a fragmentation map.

The **fragment-threshold** and **no-fragmentation** statements are mutually exclusive.

Default If you do not include this statement, the traffic in forwarding class ***class-name*** is fragmented.

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

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Configuring Fragmentation by Forwarding Class on page 709

Copyright © 2019, Juniper Networks, Inc.

forwarding-class (Interfaces)

Syntax forwarding-class *class-name*;

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

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

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

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

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

Related Documentation
- Applying Forwarding Classes to Interfaces on page 231

forwarding-class (Multifield Classifiers)

Syntax forwarding-class *class-name*;

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Set the forwarding class of incoming packets.

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
- Configuring Multifield Classifiers on page 98

Copyright © 2019, Juniper Networks, Inc.

forwarding-class (Restricted Queues)

Syntax

```
forwarding-class class-name queue queue-number;
```

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.

Description

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

Options

class-name—Name of the forwarding class.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

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

Copyright © 2019, Juniper Networks, Inc.

forwarding-class-accounting

List of Syntax

(Forwarding Class Accounting) on page 1136
(Enhanced CoS Accounting) on page 1136

(Forwarding Class Accounting)

```
forwarding-class-accounting {
  direction
}
```

(Enhanced CoS Accounting)

```
forwarding-class-accounting {
  enhanced
}
```

Hierarchy Level

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

Release Information

Forwarding class accounting statement (non-enhanced mode) introduced in Junos OS Release 13.3R3.
Enhanced CoS Accounting statement introduced in Junos OS Release 14.1.

Description

Forwarding class accounting (non-enhanced mode) is enabled for IPv4, IPv6, MPLS, Layer 2 and other packets simply by enabling the feature. This feature is supported on MX Series routers with MPCs, as well as MX80 routers.

NOTE: Forwarding class accounting (enhanced mode) is not supported in *hyper-mode (forwarding-options)*.

Both bytes and packet total are counted. For Layer 2 traffic, only bytes are counted. Flow rates are measured. Counters can be enabled in ingress, egress or in both directions. Statistics are accounted before queueing drop stage, so will include packets that might get dropped in queueing.

CoS-based interface counters (enhanced mode) are supported for IPv4 and IPv6 aggregate routes on Layer 2. This feature is supported on all MX Series routers with MPCs, as well as MX80 routers.

Both bytes and packet total are counted. Flow rates are not measured. The counters can be configured to exclude overhead bytes (such as protocol encapsulations) so end-customer packets can be differentiated from other traffic.

Disable or delete **forwarding-class-accounting** by removing the statement from the interface configuration—for example, by typing **delete interfaces *interface-name* forwarding-class-accounting**.

Copyright © 2019, Juniper Networks, Inc.

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

Related Documentation
- CoS-Based Interface Counters for IPv4 or IPv6 Aggregate on Layer 2 on page 593
- show class-of-service interface on page 1364
- *clear interfaces statistics*
- show interfaces forwarding-class-counters on page 1416

forwarding-class-default (Forwarding Policy)

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

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

Release Information
Statement introduced in Junos OS Release 14.1.
Statement introduced for QFX10000 Series switches in Junos OS Release 17.1R1.

Description
Define the next hop for traffic that does not meet any forwarding class in the next-hop map.

Options
class-name—Name of the forwarding class.

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation
- forwarding-class (Forwarding Policy) on page 1132
- *Load Balancing VPLS Non-Unicast Traffic Across Member Links of an Aggregate Interface*

Copyright © 2019, Juniper Networks, Inc.

forwarding-classes-interface-specific

Syntax

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

Hierarchy Level

[edit class-of-service]

Release Information

Statement introduced in Junos OS Release 9.6.

Description

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

Options

class-name—Name of the forwarding class.

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

queue-number—Number of the egress queue.

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

Usage Guidelines

See "Configuring a Custom Forwarding Class for Each Queue" on page 212 and "Classifying Packets by Egress Interface" on page 221.

Required Privilege Level

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

Related Documentation

- Configuring a Custom Forwarding Class for Each Queue on page 212
- Classifying Packets by Egress Interface on page 221
- output-forwarding-class-map on page 1216

Copyright © 2019, Juniper Networks, Inc.

(OBSOLETE) forwarding-classes (Class-of-Service)

Syntax

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

Hierarchy Level

[edit class-of-service]

Release Information

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

Description

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

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

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

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

Related Documentation

- Configuring a Custom Forwarding Class for Each Queue on page 212
- Forwarding Classes and Fabric Priority Queues on page 234
- Configuring Hierarchical Layer 2 Policers on IQE PICs on page 751
- Classifying Packets by Egress Interface on page 221

Copyright © 2019, Juniper Networks, Inc.

forwarding-policy

Syntax

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

Hierarchy Level

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

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced for QFX10000 Series switches in Junos OS Release 17.1R1 to support CoS-based forwarding (CBF). **[set class-of-service forwarding-policy class]** is not supported on QFX10000 Series switches.

Description

Define CoS-based forwarding policy options.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- Configuring CoS-Based Forwarding on page 224

Copyright © 2019, Juniper Networks, Inc.

fragment-threshold (Forwarding Class Maps)

Syntax fragment-threshold *bytes*;

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

Release Information Statement introduced before Junos OS Release 7.4.

Description For link services IQ (**lsq**) interfaces only, set the fragmentation threshold for an individual forwarding class.

Default If you do not include this statement, the fragmentation threshold you set at the **[edit interfaces *interface-name* unit *logical-unit-number*]** or **[edit interfaces *interface-name* mlfr-uni-nni-bundle-options]** hierarchy level is the default for all forwarding classes. If you do not set a maximum fragment size anywhere in the configuration, packets are fragmented if they exceed the smallest maximum transmission unit (MTU) of all the links in the bundle.

Options ***bytes***—Maximum size, in bytes, for multilink packet fragments. Any nonzero value must be a multiple of 64 bytes.

Range: 128 through 16,320 bytes

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

Related Documentation

- *Configuring CoS Fragmentation by Forwarding Class on LSQ Interfaces*

Copyright © 2019, Juniper Networks, Inc.

fragmentation-map

Syntax fragmentation-map *map-name*;

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

Release Information Statement introduced before Junos OS Release 7.4.

Description For AS PIC link services IQ (**lsq**) and virtual LSQ redundancy (**rlsq**) interfaces, associate a fragmentation map with a multilink PPP interface or MLFR FRF.16 DLCI.

Default If you do not include this statement, traffic in all forwarding classes is fragmented.

Options *map-name*—Name of the fragmentation map.

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

Related Documentation

- Fragmentation by Forwarding Class Overview on page 708
- Configuring Fragmentation by Forwarding Class on page 709
- Configuring Fragmentation by Forwarding Class on page 709
- Example: Configuring Fragmentation by Forwarding Class on page 712
- Configuring Drop Timeout Interval for Fragmentation by Forwarding Class on page 710
- **fragmentation-maps on page 1143**

Copyright © 2019, Juniper Networks, Inc.

fragmentation-maps

Syntax

```
fragmentation-maps {
  map-name {
    forwarding-class class-name {
      drop-timeout milliseconds;
      fragment-threshold bytes;
      multilink-class number;
      no-fragmentation;
    }
  }
}
```

Hierarchy Level

[edit class-of-service]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For Multiservices and Services PIC link services IQ (**lsq**) and virtual LSQ redundancy (**rlsq**) interfaces, define fragmentation properties for individual forwarding classes.

Default

If you do not include this statement, traffic in all forwarding classes is fragmented.

Options

map-name—Name of the fragmentation map.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

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

Related Documentation

- Fragmentation by Forwarding Class Overview on page 708
- Configuring Fragmentation by Forwarding Class on page 709
- Example: Configuring Fragmentation by Forwarding Class on page 712
- Configuring Drop Timeout Interval for Fragmentation by Forwarding Class on page 710
- **fragmentation-map on page 1142**

Copyright © 2019, Juniper Networks, Inc.

frame-mode (Dynamic Traffic Shaping)

Syntax

```
frame-mode (bytes | $junos-cos-byte-adjust | frame-mode-bytes frame-mode-bytes
  |$junos-cos-byte-adjust-frame);
```

Hierarchy Level

```
[edit dynamic-profiles profile-name class-of-service traffic-control-profiles profile-name
  overhead-accounting],
[edit class-of-service traffic-control-profiles profile-name overhead-accounting],
```

Release Information

Statement introduced in Junos OS Release 10.2.
Variable ***$junos-cos-byte-adjust-frame*** introduced in Junos OS Release 13.1.

Description

Configure the mode to shape downstream ATM traffic based as frames.

Default

The default is **frame-mode**.

Options

bytes—Byte adjustment value for the **cell-mode** or **frame-mode** shaping options.

$junos-cos-byte-adjust—Predefined variable for byte adjustment 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.

frame-mode-bytes *frame-mode-bytes*—Overhead bytes when in frame-mode. Traffic shaping is based on the number of bytes in the frame, without regard to cell encapsulation or padding overhead.

$junos-cos-byte-adjust-frame—Predefined variable for frame mode shaping. This variable can not be used when the **overhead-accounting bytes *bytes*** option is configured.

BEST PRACTICE: We recommend using the **frame-mode-bytes *frame-mode-bytes*** option rather than the **bytes** option.

Range: –120 through 124 bytes

NOTE: If you specify a value for the **bytes *bytes*** option, you cannot specify a value for either the **frame-mode-bytes** option.

Required Privilege Level

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

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- *CoS Adjustment Control Profiles Overview*
- *Configuring CoS Adjustment Control Profiles*
- *adjustment-control-profiles*
- *Configuring Dynamic Shaping Parameters to Account for Overhead in Downstream Traffic Rates*
- Bandwidth Management for Downstream Traffic in Edge Networks Overview on page 984
- egress-shaping-overhead on page 1098
- bytes on page 1059
- cell-mode on page 1062

frame-relay-de (Defining Loss Priority Maps)

Syntax

```
frame-relay-de name {
  loss-priority level code-points [ alias | bits ];
}
```

Hierarchy Level

```
[edit class-of-service loss-priority-maps]
```

Release Information Statement introduced in Junos OS Release 11.4.

Description Define a Frame Relay discard eligibility (DE) bit loss priority map.

Options ***name***—Name of the loss priority map.

loss-priority *level*—Level of the loss priority to be applied based on the specified CoS values. The loss priority level can be one of the following:

- **high**—Packet has high loss priority.
- **low**—Packet has low loss priority.
- **medium-high**—Packet has medium-high loss priority.
- **medium-low**—Packet has medium-low loss priority.

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Defining a Custom Frame Relay Loss Priority Map on page 409

Copyright © 2019, Juniper Networks, Inc.

frame-relay-de (Defining Loss Priority Rewrites)

Syntax

```
frame-relay-de name {
  loss-priority level code-point [ alias | bits ];
}
```

Hierarchy Level

```
[edit class-of-service loss-priority-rewrites]
```

Release Information Statement introduced in Junos OS Release 11.4.

Description Define a Frame Relay discard eligibility (DE) bit loss priority rewrite.

Options ***name***—Name of the loss priority rewrite.

loss-priority *level*—Level of the loss priority to be applied based on the specified CoS values. The loss priority level can be one of the following:

- **high**—Packet has high loss priority.
- **low**—Packet has low loss priority.
- **medium-high**—Packet has medium-high loss priority.
- **medium-low**—Packet has medium-low loss priority.

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Defining Custom Frame Relay Rewrite Rule on IQE PICs on page 791

Copyright © 2019, Juniper Networks, Inc.

from (Services CoS)

Syntax

```
from {
  applications [ application-name ];
  application-sets [ set-name ];
  destination-address address;
  source-address address;
}
```

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 8.1.

Description

Specify input conditions for a CoS term.

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

- Configuring CoS Rule Sets on Services PICs on page 705

Copyright © 2019, Juniper Networks, Inc.

ftp (Services CoS)

Syntax

```
ftp {
  data {
    dscp (alias | bits);
    forwarding-class class-name;
  }
}
```

Hierarchy Level

[edit services cos application-profile *profile-name* ftp]

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Set the appropriate **dscp** and **forwarding-class** value for FTP.

Default

By default, the system does not alter the DSCP or forwarding class for FTP traffic.

Required Privilege Level

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

Related Documentation

- Configuring CoS Rules on Services PICs on page 699
- sip (Application Profile) on page 1289

Copyright © 2019, Juniper Networks, Inc.

guaranteed-rate

Syntax guaranteed-rate (percent *percentage* | *rate*) <burst-size *bytes*>;

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

Release Information Statement introduced in Junos OS Release 7.6.
Option **burst-size** introduced for Enhanced Queuing (EQ) DPC interfaces in Junos OS Release 9.4.
Option **burst-size** introduced for MIC and MPC interfaces in Junos OS Release 11.4.
Option **burst-size** introduced for IQ2 and IQ2E interfaces in Junos OS Release 12.3

Description For Gigabit Ethernet IQ, Channelized IQ PICs, Multiservices and Services PICs FRF.16 LSQ interfaces, and EQ DPCs only, configure a guaranteed minimum rate. You can also configure an optional burst size for a logical interface on EQ DPCs and on IQ2 and IQ2E PICs. This can help to ensure that higher priority services do not starve lower priority services.

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 **percent** ***percentage***—For LSQ interfaces, guaranteed rate as a percentage of the available interface bandwidth.

Range: 1 through 100 percent

rate—For IQ and IQ2 interfaces, 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 6,400,000,000,000 bps

NOTE: **Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.**

burst-size ***bytes***—(Optional) Maximum burst size, in bytes.

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

Related Documentation

- Providing a Guaranteed Minimum Rate on page 285

Copyright © 2019, Juniper Networks, Inc.

- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- output-traffic-control-profile on page 1219

hierarchical-scheduler

Syntax

```
hierarchical-scheduler;
```

Hierarchy Level

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

Release Information Statement introduced in Junos OS Release 8.5.

Description On MX Series, M Series, and T Series routers with IQ2E PIC, enables the use of hierarchical schedulers.

NOTE: To enable hierarchical scheduling on MX80 and MX104 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.

Default If you do not include this statement, the interfaces on the MX Series router cannot use hierarchical interfaces.

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

Related Documentation

- Configuring Hierarchical Schedulers for CoS on page 342
- *Understanding Hierarchical CoS for Subscriber Interfaces*
- hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 1154

Copyright © 2019, Juniper Networks, Inc.

high-plp-max-threshold

Syntax high-plp-max-threshold *percent*;

Hierarchy Level [edit interfaces at-*fpc/pic/port* atm-optionslinear-red-profiles *profile-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM2 IQ interfaces only, define the drop profile fill-level for the high PLP CoS VC. When the fill level exceeds the defined percentage, all packets are dropped.

Options ***percent***—Fill-level percentage when linear random early detection (RED) is applied to cells with PLP.

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

Related Documentation

- Configuring Linear RED Profiles on ATM Interfaces on page 859
- low-plp-max-threshold on page 1202
- low-plp-threshold on page 1203
- queue-depth on page 1241

Copyright © 2019, Juniper Networks, Inc.

high-plp-threshold

Syntax high-plp-threshold *percent*;

Hierarchy Level [edit interfaces at-*fpc/pic/port* atm-optionslinear-red-profiles *profile-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM2 IQ interfaces only, define CoS VC drop profile fill-level percentage when linear RED is applied to cells with high PLP. When the fill level exceeds the defined percentage, packets with high PLP are randomly dropped by RED. This statement is mandatory.

Options *percent*—Fill-level percentage when linear RED is applied to cells with PLP.

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

Related Documentation

- Configuring Linear RED Profiles on ATM Interfaces on page 859
- high-plp-max-threshold on page 1151
- low-plp-max-threshold on page 1202
- low-plp-threshold on page 1203
- queue-depth on page 1241

Copyright © 2019, Juniper Networks, Inc.

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

Syntax

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

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 8.4.
Statement introduced before Junos OS Release 11.4 for EX Series switches.
Support for **ieee-802.1** statement introduced in Junos OS Release 12.3.
Support for distributed protocol handler traffic introduced in Junos OS Release 13.2.
Support for **protocol** statement introduced in Junos OS Release 17.3 for MX Series and PTX Series devices.

Description

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

Default

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

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- Default Routing Engine Protocol Queue Assignments on page 234
- Default DSCP and DSCP IPv6 Classifiers on page 41
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 241.
- Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic on page 597
- Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 597

Copyright © 2019, Juniper Networks, Inc.

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.
implicit-hierarchy option added 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.
Support for up to four hierarchy levels added in Junos OS Release 16.1.

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 four-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, level 3, and level 4. The **implicit-hierarchy** option is supported only on MPC/MIC subscriber interfaces and interface sets on MX Series routers.

maximum-hierarchy-levels ***number***—Specify the maximum number of hierarchical scheduling levels allowed for node scaling, from 2 through 4 levels. The default number of levels is 3. The **maximum-hierarchy-levels** option is supported on MPC/MIC or EQ DPC subscriber interfaces and interface sets on MX Series routers.

- If you set **maximum-hierarchy-levels** to **2**, interface sets are not allowed. 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, keeping the default number of hierarchy levels at 3, 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

Copyright © 2019, Juniper Networks, Inc.

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.

CAUTION: *MPC3E*, *32x10GE MPC4E*, and *2x100GE + 8x10GE MPC4E* MPCs support only two levels of scheduling hierarchy. When enabling hierarchical scheduling on these cards, you must explicitly set **maximum-hierarchy-levels** to 2.

Required Privilege Level

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

Related Documentation

- *Understanding Hierarchical CoS for Subscriber Interfaces*
- *Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links*
- Configuring Hierarchical Schedulers for CoS on page 342
- *Configuring Hierarchical CoS on a Static PPPoE Subscriber Interface*
- *Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview*

Copyright © 2019, Juniper Networks, Inc.

ieee-802.1 (Classifier on Physical Interface)

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

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

Release Information Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.

Description For ACX Series Universal Metro routers and EX Series switches, map the ieee-802.1p field of the incoming packet to the forwarding class and packet loss priority based on the specified 802.1p classifier. In the case of double tagged packets, you can configure whether to use the 802.1p of the outer or inner VLAN tag.

Options

vlan-tag inner—In the case of double tagged packets, classify based on the 802.1p of the inner VLAN tag.

vlan-tag outer—Classify based on the 802.1p of the outermost VLAN tag.

classifier-name—Name of the previously defined ieee-802.1p behavior aggregate classifier.

default—Default ieee-802.1p behavior aggregate classifier.

Required Privilege Level

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

Copyright © 2019, Juniper Networks, Inc.

ieee-802.1 (Host Outbound Traffic)

Syntax

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

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 12.3.

Description

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

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

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

Related Documentation

- Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic on page 597
- Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 597
- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388
- Configuring Rewrite Rules on page 391

Copyright © 2019, Juniper Networks, Inc.

ieee-802.1 (Rewrite Rules on Logical Interface)

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

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

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

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

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

default—The default mapping.

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

Related Documentation

- Configuring Rewrite Rules on page 391
- dscp (Rewrite Rules) on page 1093
- dscp-ipv6 (CoS Rewrite Rules) on page 1096
- exp on page 1113
- exp-push-push-push on page 1114
- exp-swap-push-push on page 1115
- ieee-802.1ad on page 1160
- inet-precedence (CoS Rewrite Rules) on page 1164
- rewrite-rules (Definition) on page 1248

Copyright © 2019, Juniper Networks, Inc.

ieee-802.1 (Rewrite Rules on Physical Interface)

Syntax ieee-802.1 (*rewrite-name* | default) ;

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

Release Information Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.

Description Apply an IEEE-802.1 rewrite rule.

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

default—The default mapping.

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

Copyright © 2019, Juniper Networks, Inc.

ieee-802.1ad

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

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

Release Information Statement introduced in Junos OS Release 9.2.

Description Apply a IEEE-802.1ad rewrite rule.

Options

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

default—The default rewrite bit mapping.

vlan-tag—The rewrite rule is applied to the **outer** or **outer-and-inner** VLAN tag.

Required Privilege Level

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

Related Documentation

- Configuring Rewrite Rules on page 391
- dscp (Rewrite Rules) on page 1093
- dscp-ipv6 (CoS Rewrite Rules) on page 1096
- exp on page 1113
- exp-push-push-push on page 1114
- exp-swap-push-push on page 1115
- ieee-802.1 (Rewrite Rules on Logical Interface) on page 1158
- inet-precedence (CoS Rewrite Rules) on page 1164
- rewrite-rules (Definition) on page 1248

Copyright © 2019, Juniper Networks, Inc.

import (Classifiers)

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

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify a default or previously defined classifier.

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

default—The default classifier mapping.

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

Related Documentation
- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36

import (Rewrite Rules)

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

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

Release Information Statement introduced before Junos OS Release 7.4.

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

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

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

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

Related Documentation
- Configuring Rewrite Rules on page 391

Copyright © 2019, Juniper Networks, Inc.

inet-precedence (Classifier on Physical Interface)

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

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

Release Information Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.

Description On ACX Series Universal Metro routers and EX Series switches, map the inet-precedence field of the incoming packet to the forwarding class and packet loss priority, based on the specified inet-precedence classifier. When no classifier is configured on the physical interface, the default ipprec-compatibility inet-precedence classifier is applied on the physical interface.

Options ***classifier-name***—Name of the previously defined inet-precedence behavior aggregate classifier.

default—Default inet-precedence behavior aggregate classifier.

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

Copyright © 2019, Juniper Networks, Inc.

inet-precedence (CoS Classifiers)

Syntax

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

Hierarchy Level

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Apply an IPv4 classifier.

Options **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.

The remaining statements are explained separately. See CLI Explorer.

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

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36

Copyright © 2019, Juniper Networks, Inc.

inet-precedence (CoS Rewrite Rules)

Syntax inet-precedence (*rewrite-name* | default) protocol (inet-both | inet-outer | mpls);

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Apply an 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.

protocol inet-both—For gr- interfaces (GRE tunnels) on MPCs, rewrite the inet-precedence CoS value to both the inner and outer header for Unicast/Multicast IPv4 traffic. The first three bits of the CoS value are rewritten and the final six bits are taken from the incoming CoS value.

protocol inet-outer—For gr- interfaces on MPCs, rewrite the inet-precedence CoS value to the outer header for Unicast/Multicast IPv4 traffic. The first three bits of the CoS value are rewritten and the final six bits are taken from the incoming CoS value.

protocol mpls—(Optional for ingress MPLS tunnel nodes) For interfaces on MX Series routers or hosted on Enhanced III FPCs in M120 or M320 routers only, rewrite the MPLS EXP bits in the MPLS header independently of the IPv4 DSCP value for IPv4 packets entering an MPLS tunnel.

Required Privilege Level

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

Related Documentation

- Configuring Rewrite Rules on page 391
- Applying Rewrite Rules to Output Logical Interfaces on page 401
- **protocol (Rewrite Rules) on page 1237**
- **rewrite-rules (Definition) on page 1248**

Copyright © 2019, Juniper Networks, Inc.

inet-precedence (Rewrite Rules on Physical Interface)

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

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

Release Information Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.

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.

Copyright © 2019, Juniper Networks, Inc.

inet6-precedence (CoS Rewrite Rules)

Syntax

```
inet6-precedence rewrite-name {
  protocol mpls;
}
```

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 18.1R1 for MX Series routers with MPCs.

Description

Apply an IPv6 precedence rule to rewrite the first three bits of the IPv6 DSCP value.

Options

rewrite-name—Name of a **rewrite-rules** mapping configured at the **[edit class-of-service rewrite-rules inet6-precedence]** hierarchy level.

protocol mpls—(Optional for ingress MPLS tunnel nodes) Rewrite the DSCP value for IPv6 packets entering the MPLS tunnel.

Required Privilege Level

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

Related Documentation

- Configuring Rewrite Rules on page 391
- Applying Rewrite Rules to Output Logical Interfaces on page 401
- **protocol (Rewrite Rules) on page 1237**
- **rewrite-rules (Definition) on page 1248**

Copyright © 2019, Juniper Networks, Inc.

ingress-policer-overhead

Syntax ingress-policer-overhead *bytes*;

Hierarchy Level [edit chassis fpc *slot-number* pic *pic-number*]

Release Information Statement introduced before Junos OS Release 11.1.
Statement introduced in Junos OS Release 15.1X49-D30 for vSRX.

Description Add the configured number of bytes to the length of a packet entering the interface.

Configure a policer overhead to control the rate of traffic received on an interface. Use this feature to help prevent denial-of-service (DoS) attacks or to enforce traffic rates to conform to the service-level agreement (SLA). When you configure a policer overhead, the configured policer overhead value (bytes) is added to the length of the final Ethernet frame. This calculated length of frame is used to determine the policer or the rate-limiting action.

Traffic policing combines the configured policy bandwidth limits and the burst size to determine how to meter the incoming traffic. If you configure a policer overhead on an interface, Junos OS adds those bytes to the length of incoming Ethernet frames. This added overhead fills each frame closer to the burst size, allowing you to control the rate of traffic received on an interface.

You can configure the policer overhead to rate-limit queues and Layer 2 and Layer 3 policers, for standalone (SA) and high-avalability (HA) deployments. The policer overhead and the shaping overhead can be configured simultaneously on an interface.

NOTE: vSRX supports policer overhead on Layer 3 policers only.

The policer overhead applies to all interfaces on the PIC. In the following example, Junos OS adds 10 bytes of overhead to all incoming Ethernet frames on ports ge-0/0/0 through ge-0/0/4.

```
set chassis fpc 0 pic 0 ingress-policer-overhead 10
```

NOTE: vSRX only supports fpc 0 pic 0. When you commit the ingress-policer-overhead statement, the vSRX takes the PIC offline and then back online.

Copyright © 2019, Juniper Networks, Inc.

You need to craft the policer overhead size to match your network traffic. A value that is too low will have minimal impact on traffic bursts. A value that is too high will rate-limit too much of your incoming traffic.

In this example, the policer overhead of 255 bytes is configured for ge-0/0/0 through ge-0/0/4. The firewall policer is configured to discard traffic when the burst size is over 1500 bytes. This policer is applied to ge-0/0/0 and ge 0/0/1. Junos OS adds 255 bytes to every Ethernet frame that comes into the configured ports. If, during a burst of traffic, the combined length of incoming frames and the overhead bytes exceeds 1500 bytes, the policer starts to discard further incoming traffic.

```
set chassis fpc 0 pic 0 ingress-policer-overhead 255
set interfaces ge-0/0/0 unit 0 family inet policer input overhead_policer
set interfaces ge-0/0/0 unit 0 family inet address 10.9.1.2/24
set interfaces ge-0/0/1 unit 0 family inet policer input overhead_policer
set interfaces ge-0/0/1 unit 0 family inet address 10.9.2.2/24
set firewall policer overhead_policer if-exceeding bandwidth-limit 32k
set firewall policer overhead_policer if-exceeding burst-size-limit 1500
set firewall policer overhead_policer then discard
```

Options

bytes—Number of bytes added to a frame entering an interface.

Range: 0–255 bytes

Default: 0

```
[edit chassis fpc 0 pic 0]
user@host# set ingress-policer-overhead 10;
```

Required Privilege Level

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

Related Documentation

- ingress-shaping-overhead on page 1170
- Policer Overhead to Account for Rate Shaping Overview on page 195
- Example: Configuring Policer Overhead to Account for Rate Shaping on page 196
- Configuring a Policer Overhead on page 817
- CoS on Enhanced IQ2 PICs Overview on page 794

Copyright © 2019, Juniper Networks, Inc.

ingress-queuing-filter

Syntax

```
ingress-queuing-filter filter-name;
```

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 16.1 for MX Series routers with MPCs.

Description

Use the **ingress-queuing-filter** statement to set the packet loss priority and forwarding class for the packet, or drop the packet prior to input queue selection. This assists in traffic shaping.

The **ingress-queuing-filter** statement is available only for the following protocol families: **bridge, ccc, inet, inet6, mpls,** and **vpls**.

ingress-queuing-filter takes ***filter-name*** as an argument. The named filter is a normal firewall filter that must be configured with at least one of the following actions: **accept, discard, forwarding-class,** and **loss-priority**.

Required Privilege Level

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

Related Documentation

- Example: Configuring a Filter for Use as an Ingress Queuing Filter on page 945

Copyright © 2019, Juniper Networks, Inc.

ingress-shaping-overhead

Syntax ingress-shaping-overhead *number*;

Hierarchy Level [edit chassis fpc *slot-number* pic *pic-number* traffic-manager],
[edit chassis lcc *number* fpc *slot-number* pic *pic-number* traffic-manager]

Release Information Statement introduced in Junos OS Release 8.3.

Description Number of bytes to add to packet to determine shaped session packet length.

NOTE: Although the ingress queuing filter can be used with EX9200 switches and T-Series routers as well as MX-Series routers, it is used only on those MX Series routers that have MPCs. An error is generated at commit if the ingress queuing filter is applied to an interface on any other type of port concentrator.

Options *number*—When L2TP session shaping is configured, the number of CoS shaping overhead bytes to add to the packets on the ingress side of the L2TP tunnel to determine the shaped session packet length.

When session shaping is not configured and traffic management (queuing and scheduling) is configured on the ingress side, the number of CoS shaping overhead bytes to add to the packets on the ingress interface.

Range: –63 through 192

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

Related Documentation

- Configuring CoS for L2TP Tunnels on ATM Interfaces on page 848
- egress-shaping-overhead on page 1098
- mode (Layer 2 Tunneling Protocol Shaping) on page 1211
- traffic-manager on page 1306

Copyright © 2019, Juniper Networks, Inc.

input-excess-bandwidth-share

Syntax

```
input-excess-bandwidth-share (proportional value | equal);
```

Hierarchy Level

[edit class-of-service interfaces *interface-name*],
[edit class-of-service interfaces interface-set *interface-set-name*]

Release Information

Statement introduced in Junos OS Release 9.0.

Description

Determines the method of sharing excess bandwidth on the ingress interface in a hierarchical scheduler environment. If you do not include this statement, the node shares excess bandwidth proportionally at 32.64 Mbps.

Options

proportional *value*—(Default) Share ingress excess bandwidth proportionally (default value is 32.64 Mbps).

equal—Share ingress excess bandwidth equally.

Required Privilege Level

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

Related Documentation

- Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs on page 920

Copyright © 2019, Juniper Networks, Inc.

input-policer

Syntax

```
input-policer policer-name;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number layer2-policer]
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number
  layer2-policer]
```

Release Information

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

Description

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

Options

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

Usage Guidelines

See "Applying Layer 2 Policers to Gigabit Ethernet Interfaces" on page 1033.

Required Privilege Level

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

Related Documentation

- *Two-Color and Three-Color Policers at Layer 2*
- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 1033
- *Configuring a Gigabit Ethernet Policer*
- input-three-color on page 1176
- layer2-policer on page 1187
- logical-interface-policer on page 1192
- output-policer on page 1217
- output-three-color on page 1218

Copyright © 2019, Juniper Networks, Inc.

input-scheduler-map

Syntax

```
input-scheduler-map map-name;
```

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 7.6.

Description

Associate a scheduler map with a physical or logical input interface. The **input-scheduler-map** and **input-traffic-control-profile** statements are mutually exclusive at the same hierarchy level.

input-scheduler-map is supported on the following Ethernet interfaces:

- IQ2 and IQ2E PICs
- DPCs and MPCs that support Enhanced Queuing (Q/EQ)
- MX80 with support for per-VLAN queuing

NOTE:

For an Enhanced Queuing (EQ) DPC on an MX Series router, CoS queuing and scheduling are enabled on the egress side but disabled on the ingress side by default. To enable ingress CoS on the EQ DPC, you must configure the traffic-manager statement with ingress-and-egress mode:

```
[edit chassis fpc slot-number pic pic-number]
traffic-manager mode ingress-and-egress;
```

Options

map-name—Name of scheduler map that you define at the **[edit class-of-service scheduler-maps]** hierarchy level.

default—The default scheduler mapping.

Required Privilege Level

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

Related Documentation

- Configuring an Input Scheduler on an Interface on page 263
- Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs on page 920
- input-traffic-control-profile on page 1177

Copyright © 2019, Juniper Networks, Inc.

input-shaping-rate (Logical Interface)

Syntax input-shaping-rate (percent *percentage* | *rate*);

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

Release Information Statement introduced in Junos OS Release 7.6.

Description For Gigabit Ethernet IQ2, Enhanced Queuing DPC, MIC, and MPC interfaces, configure input traffic shaping by specifying the amount of bandwidth to be allocated to the logical interface. You can configure hierarchical shaping, meaning you can apply an input shaping rate to both the physical interface and the logical interface.

Default If you do not include this statement, logical interfaces share a default scheduler. This scheduler has a committed information rate (CIR) that equals 0. (The CIR is the guaranteed rate.) The default scheduler has a peak information rate (PIR) that equals the physical interface shaping rate.

Options **percent *percentage***—Shaping rate as a percentage of the available interface bandwidth.

Range: 0 through 100 percent

rate—Peak 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 6,400,000,000,000 bps

NOTE: Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.

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

Related Documentation

- Configuring Ingress Hierarchical CoS on MIC and MPC Interfaces on page 975
- Configuring Input Shaping Rates for Both Physical and Logical Interfaces on page 326
- Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs on page 920
- input-traffic-control-profile on page 1177

Copyright © 2019, Juniper Networks, Inc.

input-shaping-rate (Physical Interface)

Syntax input-shaping-rate *rate*;

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

Release Information Statement introduced in Junos OS Release 7.6.

Description For Gigabit Ethernet IQ2, Enhanced Queuing DPC, MIC, and MPC interfaces, configure input traffic shaping by specifying the amount of bandwidth to be allocated to the physical interface. You can configure hierarchical shaping, meaning you can apply an input shaping rate to both the physical interface and the logical interface.

Options ***rate***—Peak 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 6,400,000,000,000 bps

NOTE: **Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.**

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

Related Documentation

- Configuring Input Shaping Rates for Both Physical and Logical Interfaces on page 326
- Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs on page 920
- **input-traffic-control-profile on page 1177**

Copyright © 2019, Juniper Networks, Inc.

input-three-color

Syntax

```
input-three-color policer-name;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number layer2-policer]
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number
  layer2-policer]
```

Release Information

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

Description

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

Options

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

Usage Guidelines

See "Applying Layer 2 Policers to Gigabit Ethernet Interfaces" on page 1033.

Required Privilege Level

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

Related Documentation

- *Two-Color and Three-Color Policers at Layer 2*
- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 1033
- *Configuring a Gigabit Ethernet Policer*
- input-policer on page 1172
- layer2-policer on page 1187
- logical-interface-policer on page 1192
- output-policer on page 1217
- output-three-color on page 1218

Copyright © 2019, Juniper Networks, Inc.

input-traffic-control-profile

Syntax

```
input-traffic-control-profile profile-name shared-instance instance-name;
```

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 7.6.

Description

For Gigabit Ethernet IQ2 and IQ2E PIC, Enhanced Queuing DPC , MIC, and MPC interfaces, apply an input traffic scheduling and shaping profile to the logical interface. The **input-traffic-control-profile** and **input-scheduler-map** statements are mutually exclusive at the same hierarchy level.

NOTE: The **shared-instance** statement applies only to Gigabit Ethernet IQ2 and IQ2E PICs.

NOTE:

For an Enhanced Queuing (EQ) DPC on an MX Series router, CoS queuing and scheduling are enabled on the egress side but disabled on the ingress side by default. To enable ingress CoS on the EQ DPC, you must configure the traffic-manager statement with ingress-and-egress mode:

```
[edit chassis fpc slot-number pic pic-number]
traffic-manager mode ingress-and-egress;
```

Options

profile-name—Name of the traffic-control profile to be applied to this interface.

instance-name—Name of the shared scheduler and shaper to be applied to this interface.

Required Privilege Level

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

Related Documentation

- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs on page 920
- input-shaping-rate (Logical Interface) on page 1174
- input-scheduler-map on page 1173
- traffic-control-profiles on page 1305

Copyright © 2019, Juniper Networks, Inc.

input-traffic-control-profile-remaining

Syntax

```
input-traffic-control-profile-remaining profile-name;
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name],
[edit class-of-service interfaces interface-name interface-set interface-set-name]
```

Release Information Statement introduced in Junos OS Release 9.0.

Description For Enhanced Queuing DPC, MICs, or MPC interfaces on MX Series routers, or for IQ2E PICs interfaces on M Series and T Series router, apply an input traffic scheduling and shaping profile for the remaining traffic to the logical interface or interface set.

Options *profile-name*—Name of the traffic-control profile for the remaining traffic to be applied to this interface or interface set.

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

Related Documentation

- Configuring Ingress Hierarchical CoS on Enhanced Queuing DPCs on page 920
- input-traffic-control-profile on page 1177

Copyright © 2019, Juniper Networks, Inc.

interface-set (Ethernet Interfaces)

Syntax

```
interface-set interface-set-name {
  interface ethernet-interface-name {
    (unit unit-number | vlan-tags-outer vlan-tag);
  }
}
```

Hierarchy Level

```
[edit interfaces]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

The set of interfaces used to configure hierarchical CoS schedulers on Ethernet interfaces on the MX Series router and IQ2E PIC on M Series and T Series routers.

The remaining statements are described separately.

Required Privilege Level

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

Related Documentation

- interface-set (Hierarchical Schedulers) on page 1180

Copyright © 2019, Juniper Networks, Inc.

interface-set (Hierarchical Schedulers)

Syntax

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

Hierarchy Level

```
[edit class-of-service interfaces]
```

Release Information

Statement introduced in Junos OS Release 8.5.

Description

For Enhanced Queuing DPC, MIC, or MPC interfaces on MX Series routers, or for IQ2E PIC interfaces on M Series routers, configure hierarchical schedulers for an interface set.

Options

interface-set-name—Name of the interface set.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Interface Sets on page 264
- Configuring Hierarchical Schedulers for CoS on page 342

Copyright © 2019, Juniper Networks, Inc.

interfaces (CoS)

Syntax

```
interfaces {
  interface-name {
    classifiers{
      dscp(classifier-name | default) {
    }
      ieee-802.1 (classifier-name | default)  vlan-tag (inner | outer | classifier-name);
      inet-precedence (rewrite-name | default);
    }
    input-scheduler-map map-name;
    input-shaping-rate rate;
    irb {
      unit logical-unit-number {
        classifiers {
          type (classifier-name | default);
          no-default;
        }
        rewrite-rules {
          dscp (rewrite-name | default);
          dscp-ipv6 (rewrite-name | default);
          exp (rewrite-name | default)protocol protocol-types;
          ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
          inet-precedence (rewrite-name | default);
        }
      }
    }
    member-link-scheduler (replicate | scale);
    rewrite-rules {
      dscp (rewrite-name | default);
      ieee-802.1 (rewrite-name | default) vlan-tag (outer);
      inet-precedence (rewrite-name | default);
      }
    }
    scheduler-map map-name;
    scheduler-map-chassis map-name;
    shaping-rate rate;
    unit logical-unit-number {
      classifiers {
        type (classifier-name | default) family (mpls | inet);
      }
      forwarding-class class-name;
      fragmentation-map map-name;
      input-shaping-rate (percent percentage | rate);
      input-traffic-control-profile profile-name shared-instance instance-name;
      output-traffic-control-profile profile-name shared-instance instance-name;
      per-session-scheduler;
      policy-map policy-map-name;
      rewrite-rules {
        dscp (rewrite-name | default);
        dscp-ipv6 (rewrite-name | default);
        exp (rewrite-name | default) protocol protocol-types;
        exp-push-push-push default;
```

Copyright © 2019, Juniper Networks, Inc.

```
        exp-swap-push-push default;
        ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
        inet-precedence (rewrite-name | default);
      }
      scheduler-map map-name;
      shaping-rate rate;
      translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 | to-exp-from-exp
        | to-inet-precedence-from-inet-precedence) table-name;
    }
  }
  interface-set interface-set-name {
    excess-bandwidth-share;
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.
Interface-set level added in Junos OS Release 8.5.

Description Configure interface-specific CoS properties for incoming packets.

Options The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Configuring Rewrite Rules on page 391

Copyright © 2019, Juniper Networks, Inc.

internal-node

Syntax internal-node;

Hierarchy Level [edit class-of-service interfaces interface-set *interface-set-name*]

Release Information Statement introduced in Junos OS Release 8.5.

Description The statement is used to raise the interface set without children to the same level as the other configured interface sets with children, allowing them to compete for the same set of resources.

Default If you do not include this statement, the node is internal only if its children have a traffic control profile configured.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Internal Scheduler Nodes on page 268

Copyright © 2019, Juniper Networks, Inc.

interpolate

Syntax

```
interpolate {
  drop-probability [values];
  fill-level [values];
}
```

Hierarchy Level

[edit class-of-service drop-profiles *profile-name*]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS 11.4 for EX Series switches.

Description

Specify values for interpolating relationship between queue fill level and drop probability.

On EX Series switches, this statement is supported only on EX8200 standalone switches and EX8200 Virtual Chassis.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- See Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 355.

Copyright © 2019, Juniper Networks, Inc.

iq-policing-filter

Syntax iq-policing-filter *filter-name*;

Hierarchy Level [edit interfaces *interface-name* unit *unit-number* family *bridge*|*inet*|*vpls*]

Release Information Statement introduced in Junos OS Release 18.1R1 for MX Series routers with MPCs that support ingress queuing.

Description For MPCs that support ingress queuing, attaches a filter to an interface that polices traffic before assigning to ingress queues. This enables rate limiting of traffic before the traffic is assigned to ingress queues.

The **iq-policing-filter** statement is available for the following protocol families: **bridge**, **inet**, and **vpls**.

Options **filter-name**—Name of the **iq-policing-filter** filter

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

irb

Syntax

```
irb {
  unit logical-unit-number {
    classifiers {
      type (classifier-name | default);
    }
    rewrite-rules {
      dscp (rewrite-name | default);
      dscp-ipv6 (rewrite-name | default);
      exp (rewrite-name | default)protocol protocol-types;
      ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
      inet-precedence (rewrite-name | default);
    }
  }
}
```

Hierarchy Level

[edit class-of-service interfaces]

Release Information

Statement introduced in Junos OS Release 8.4.

Description

On the MX Series routers and EX Series switches, you can apply classifiers or rewrite rules to an integrated bridging and routing (IRB) interface. All types of classifiers and rewrite rules are allowed. These classifiers and rewrite rules are independent of others configured on the MX Series router and on EX Series switches.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- CoS Features and Limitations on MX Series Routers on page 568

Copyright © 2019, Juniper Networks, Inc.

layer2-policer

Syntax

```
layer2-policer {
  input-policer policer-name;
  input-three-color policer-name;
  output-policer policer-name;
  output-three-color policer-name;
}
```

Hierarchy Level

[edit interfaces *interface-name* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*],

Release Information

Statement introduced in Junos OS Release 8.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

For 1-Gigabit Ethernet and 10-Gigabit Ethernet IQ2 and IQ2-E interfaces on M Series, MX Series, and T Series routers, and for aggregated Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet interfaces on EX Series switches, apply Layer 2 logical interface policers. The following policers are supported:

- Two-color
- Single-rate tricolor marking (srTCM)
- Two-rate tricolor marking (trTCM)

Two-color and tricolor policers are configured at the **[edit firewall]** hierarchy level.

Options

input-policer *policer-name*—Two-color input policer to associate with the interface. This statement is mutually exclusive with the **input-three-color** statement.

input-three-color *policer-name*—Tricolor input policer to associate with the interface. This statement is mutually exclusive with the **input-policer** statement.

output-policer *policer-name*—Two-color output policer to associate with the interface. This statement is mutually exclusive with the **output-three-color** statement.

output-three-color *policer-name*—Tricolor output policer to associate with the interface. This statement is mutually exclusive with the **output-policer** statement.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 1033
- *Configuring Gigabit Ethernet Two-Color and Tricolor Policers*

Copyright © 2019, Juniper Networks, Inc.

linear-red-profile

Syntax

```
linear-red-profile profile-name;
```

Hierarchy Level

```
[edit interfaces at-fpc/pic/port atm-options scheduler-maps map-name forwarding-class
  class-name]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM2 IQ interfaces only, assign a linear RED profile to a specified forwarding class. To define the linear RED profiles, include the linear-red-profiles statement at the **[edit interfaces at-*fpc/pic/port* atm-options]** hierarchy level.

Default If you do not include either the **epd-threshold** or the **linear-red-profile** statement in the forwarding class configuration, the Junos OS uses an EPD threshold based on the available bandwidth and other parameters.

Options ***profile-name***—Name of the linear RED profile.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Configuring an ATM Scheduler Map*
- linear-red-profiles on page 1189
- Applying Scheduler Maps to ATM Interfaces on page 853
- epd-threshold on page 1104

Copyright © 2019, Juniper Networks, Inc.

linear-red-profiles

Syntax

```
linear-red-profiles profile-name {
  high-plp-threshold percent;
  low-plp-threshold percent;
  queue-depth cells;
}
```

Hierarchy Level

```
[edit interfaces at-fpc/pic/port atm-options]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For ATM2 IQ interfaces only, define CoS virtual circuit drop profiles for RED. When a packet arrives, RED checks the queue fill level. If the fill level corresponds to a nonzero drop probability, the RED algorithm determines whether to drop the arriving packet.

Options

profile-name—Name of the drop profile.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *ATM2 IQ VC Tunnel CoS Components Overview*
- Configuring Linear RED Profiles on ATM Interfaces on page 859

Copyright © 2019, Juniper Networks, Inc.

logical-bandwidth-policer

Syntax

```
logical-bandwidth-policer;
```

Hierarchy Level

[edit dynamic-profiles *profile-name* firewall policer *policer-name*],
[edit firewall policer *policer-name*],
[edit logical-systems *logical-system-name* firewall policer *policer-name*]

Release Information

Statement introduced in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
Support at the **[edit dynamic-profiles ... policer *policer-name*]** hierarchy level introduced in Junos OS Release 11.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

For a policer with a bandwidth limit configured as a percentage (using the **bandwidth-percent** statement), specify that the percentage be based on the shaping rate defined on the logical interface, rather than on the media rate of the physical interface.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Bandwidth Policers*
- Configuring Policers Based on Logical Interface Bandwidth on page 122
- **bandwidth-percent** statement
- **interface-specific** statement

Copyright © 2019, Juniper Networks, Inc.

logical-interface-aggregate-statistics

Syntax logical-interface-aggregate-statistics;

Hierarchy Level [edit class-of-service interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 18.2R1.

Description Enable the collection of aggregate queue statistics for all underlying logical interfaces on a particular physical or aggregate (for example, ae0) interface. By default to preserve memory resources, aggregate queue statistics are not collected for underlying logical interfaces (Level 2 interfaces); queue statistics are only collected for the upper-level logical interfaces (Level 3 and above) and the physical interface (Level 1). The command **show interfaces queue *interface-name*** shows all zeroes for underlying logical interfaces by default.

CAUTION: To enable a hierarchical CoS feature on a physical interface with an already active interface hierarchy:

1. *deactivate* class-of-service on all hierarchy levels of the physical interface.
2. Enable the hierarchical CoS setting on the physical interface.
3. Re-*activate* class-of-service on all hierarchy levels of the physical interface.

Required Privilege Level interface

Related Documentation

- show interfaces queue on page 1421
- Hierarchical Class of Service Overview on page 333
- show class-of-service scheduler-hierarchy interface on page 1404

Copyright © 2019, Juniper Networks, Inc.

logical-interface-policer

Syntax logical-interface-policer;

Hierarchy Level
[edit dynamic-profiles *profile-name* firewall policer *policer-name*],
[edit dynamic-profiles *profile-name* firewall three-color-policer *name*],
[edit firewall atm-policer*atm-policer-name*],
[edit firewall policer *policer-name*],
[edit firewall policer *policer-template-name*],
[edit firewall three-color-policer *policer-name*],
[edit logical-systems *logical-system-name* firewall policer *policer-name*],
[edit logical-systems *logical-system-name* firewall three-color-policer *name*]

Release Information
Statement introduced before Junos OS Release 7.4.
Support at the **[edit firewall three-color-policer *policer-name*]** hierarchy level introduced in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
Support at the **[edit dynamic-profiles ... policer *policer-name*]** and **[edit dynamic-profiles ... three-color-policer *name*]**hierarchy levels introduced in Junos OS Release 11.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.
Support for PTX series routers with third-generation FPCs added in Junos OS Release 18.3R1.

Description
Configure a logical interface policer. For PTX series routers running Junos OS Release 18.3R1 or later, you can use this command to configure separate firewall filters for different family address types (IPv4 and IPv6) that share the same interface, and configure the same policer as an action for the filter.

To configure the aggregate policer, configure the firewall policer you want to use as **logical-interface-policer**. And at the **firewall family *family-name* filter *filter-name*** hierarchy level where you will reference the policer, make the policer an **interface-specific** firewall filter action.

The sample configuration shows the relationship.

```
firewall {
    policer Shared_Policer {
      logical-interface-policer;
        if-exceeding {
            bandwidth-limit 100m;
            burst-size-limit 500k;
        }
        then {
            discard;
        }
    }
}
```


Copyright © 2019, Juniper Networks, Inc.

```
family inet {
    filter filter_name{
       interface-specific;
        term term_name {
            then {
                policer Shared_Policer;
                count cinet;
            }
        }
    }
}
```

NOTE: Starting in Junos OS Release 12.2R2, on T Series Core Routers only, you can configure an MPLS LSP policer for a specific LSP to be shared across different protocol family types. You must include the **logical-interface-policer** statement to do so.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- *Two-Color and Three-Color Logical Interface Policers*
- *Traffic Policer Types*
- Configuring and Applying Tricolor Marking Policers on page 175
- **action on page 1045**
- *Configuring Gigabit Ethernet Two-Color and Tricolor Policers*
- *action*

Copyright © 2019, Juniper Networks, Inc.

loss-priority (BA Classifiers)

Syntax

```
loss-priority level;
```

Hierarchy Level

```
[edit class-of-service classifiers type classifier-name forwarding-class class-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.

Description

Specify packet loss priority value for a specific set of code-point aliases and bit patterns.

Options

level can be one of the following:

- **high**—Packet has high loss priority.
- **medium-high**—Packet has medium-high loss priority.
- **medium-low**—Packet has medium-low loss priority.
- **low**—Packet has low loss priority.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Configuring and Applying Tricolor Marking Policers on page 175

loss-priority (Firewall Filter)

Syntax

```
loss-priority (high | low | medium-high | medium-low);
```

Hierarchy Level

```
[edit firewall family family-name filter filter-name term term-name then]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Set the loss priority of incoming packets.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Multifield Classifiers on page 98

Copyright © 2019, Juniper Networks, Inc.

loss-priority (Rewrite Rules)

Syntax

```
loss-priority level;
```

Hierarchy Level

```
[edit class-of-service rewrite-rules type rewrite-name forwarding-class class-name]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify a loss priority to which to apply a rewrite rule. The rewrite rule sets the code-point aliases and bit patterns for a specific forwarding class and packet loss priority (PLP). The inputs for the map are the forwarding class and the PLP. The output of the map is the code-point alias or bit pattern.

Options *level* can be one of the following:

- **high**—The rewrite rule applies to packets with high loss priority.
- **low**—The rewrite rule applies to packets with low loss priority.
- **medium-high**—The rewrite rule applies to packets with medium-high loss priority.
- **medium-low**—The rewrite rule applies to packets with medium-low loss priority.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Rewrite Rules on page 391
- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Configuring and Applying Tricolor Marking Policers on page 175

Copyright © 2019, Juniper Networks, Inc.

loss-priority (Scheduler Drop Profiles)

Syntax

```
loss-priority (any | high | low | medium-high | medium-low);
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name drop-profile-map]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.

Description

Specify a loss priority to which to apply a drop profile. The drop profile map sets the drop profile for a specific PLP and protocol type. The inputs for the map are the PLP designation and the protocol type. The output is the drop profile.

Options

any—The drop profile applies to packets with any PLP.

high—The drop profile applies to packets with high PLP.

low—The drop profile applies to packets with low PLP.

medium-high—The drop profile applies to packets with medium-high PLP.

medium-low—The drop profile applies to packets with medium-low PLP.

NOTE: On ACX Series Routers, if you configure the protocol as tcp, then the loss-priority (any | high | low | medium-high) values are supported. If you configure the protocol with either non-tcp or any option, then irrespective of the loss-priority value, only one drop-profile can be specified.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Default Schedulers Overview on page 257
- Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 359
- Configuring Schedulers for Priority Scheduling on page 330
- Configuring and Applying Tricolor Marking Policers on page 175
- **protocol (Schedulers) on page 1238**

Copyright © 2019, Juniper Networks, Inc.

loss-priority (Simple Firewall Filter)

Syntax

```
loss-priority (high | low | medium);
```

Hierarchy Level

[edit firewall family *family-name* simple-filter *filter-name* term *term-name* then]

Release Information

Statement introduced in Junos OS Release 7.6.

Description

Set the loss priority of incoming packets.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Multifield Classifiers on page 98

Copyright © 2019, Juniper Networks, Inc.

loss-priority-maps

Syntax

```
loss-priority-maps {
  frame-relay-de rewrite-name {
    loss-priority level {
    code-points [ aliases] [ bit-patterns ];
    }
  }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information Statement introduced in JUNOS Release 11.4.

Description Map the loss priority of incoming packets based on the CoS values.

Options **frame-relay-de *rewrite-name***—Name of the Frame Relay DE bit loss priority map.

loss-priority *level*—The loss priority level can be one of the following:

- **high**—Packet has high loss priority.
- **low**—Packet has low loss priority.
- **medium-high**—Packet has medium-high loss priority.
- **medium-low**—Packet has medium-low loss priority.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface on page 565

Copyright © 2019, Juniper Networks, Inc.

loss-priority-maps (Assigning to an Interface)

Syntax

```
loss-priority-maps {
  frame-relay-de (loss-priority-rewrite-name | default);
}
```

Hierarchy Level

[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]

Release Information

Statement introduced in JUNOS Release 11.4.

Description

Assign the loss priority map to a logical interface.

Options

default—Apply the default loss priority map. The default map includes the following configuration:

```
loss-priority low code-point 0;
loss-priority high code-point 1;
```

map-name—Name of loss priority map to be applied.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface on page 565
- unit on page 1318

Copyright © 2019, Juniper Networks, Inc.

loss-priority-rewrites

Syntax

```
loss-priority-rewrites {
  frame-relay-de rewrite-name {
    loss-priority level {
    code-points [ bit-patterns ];
    }
  }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.4.

Description

Specify the Frame Relay discard eligibility (DE) bit rewrite rule on the enhanced IQ PIC.

Options

frame-relay-de *rewrite-name*—Name of the Frame Relay DE bit loss priority rewrite rule.

loss-priority *level*—The loss priority level can be one of the following:

- **high**—Packet has high loss priority.
- **low**—Packet has low loss priority.
- **medium-high**—Packet has medium-high loss priority.
- **medium-low**—Packet has medium-low loss priority.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Defining Custom Frame Relay Rewrite Rule on IQE PICs on page 791

Copyright © 2019, Juniper Networks, Inc.

loss-priority-rewrites (Assigning to an Interface)

Syntax

```
loss-priority-rewrites {
  frame-relay-de (loss-priority-rewrite-name | default);
}
```

Hierarchy Level

[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]

Release Information

Statement introduced in Junos OS Release 11.4.

Description

Associate the loss priority rewrites to an outgoing packet.

Options

loss-priority-rewrite-name—Name of the loss priority rewrite to be applied to an interface.

default—Default loss priority rewrite to be applied to an interface.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Defining Custom Frame Relay Rewrite Rule on IQE PICs on page 791
- Assigning Default Frame Relay Rewrite Rule to IQE PICs on page 790

Copyright © 2019, Juniper Networks, Inc.

low-plp-max-threshold

Syntax low-plp-max-threshold *percent*;

Hierarchy Level [edit interfaces at-*fpc/pic/port* atm-options linear-red-profiles *profile-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM2 IQ interfaces only, define the drop profile fill-level for the low PLP CoS VC. When the fill level exceeds the defined percentage, all packets are dropped.

Options ***percent***—Fill-level percentage when linear RED is applied to cells with PLP.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *ATM2 IQ VC Tunnel CoS Components Overview*
- *high-plp-max-threshold*
- low-plp-threshold on page 1203
- Configuring Linear RED Profiles on ATM Interfaces on page 859
- high-plp-max-threshold on page 1151
- queue-depth on page 1241

Copyright © 2019, Juniper Networks, Inc.

low-plp-threshold

Syntax low-plp-threshold *percent*;

Hierarchy Level [edit interfaces at-*fpc/pic/port* atm-options linear-red-profiles *profile-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM2 IQ interfaces only, define the CoS VC drop profile fill-level percentage when linear RED is applied to cells with low PLP. When the fill level exceeds the defined percentage, packets with low PLP are randomly dropped by RED. This statement is mandatory.

Options ***percent***—Fill-level percentage when linear RED is applied to cells 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

- *ATM2 IQ VC Tunnel CoS Components Overview*
- *high-plp-max-threshold*
- *high-plp-threshold*
- Configuring Linear RED Profiles on ATM Interfaces on page 859
- high-plp-max-threshold on page 1151
- high-plp-threshold on page 1152
- low-plp-max-threshold on page 1202
- queue-depth on page 1241

Copyright © 2019, Juniper Networks, Inc.

lsp-next-hop (CoS-Based Forwarding)

Syntax

```
lsp-next-hop [ lsp-regular-expression ];
```

Hierarchy Level

```
[edit class-of-service forwarding-policy next-hop-map map-name forwarding-class
  class-name]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the LSP regular expression to which to map forwarded traffic.

Options ***lsp-regular-expression***—Next-hop LSP label.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS-Based Forwarding on page 224

match-direction (Services CoS)

Syntax

```
match-direction (input | output | input-output);
```

Hierarchy Level

```
[edit services cos rule rule-name]
```

Release Information Statement introduced in Junos OS Release 8.1.

Description Specify the direction in which the rule match is applied.

Options **input**—Apply the rule match on the input side of the interface.

output—Apply the rule match on the output side of the interface.

input-output—Apply the rule match bidirectionally.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS Rules on Services PICs on page 699

Copyright © 2019, Juniper Networks, Inc.

max-burst-size

Syntax max-burst-size *max-burst-size*;

Hierarchy Level [edit class-of-service traffic-control-profiles *traffic-control-profile-name*],
[edit firewall atm-policer *atm-policer-name*]

Release Information Statement introduced in Junos OS Release 12.1.

Description (MX Series routers) Define ATM maximum burst size on ATM MICs in cells.

Options *cells*—ATM maximum burst size in cells.

Range: 1 through 4000 cells

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *show class-of-service traffic-control-profile*

Copyright © 2019, Juniper Networks, Inc.

max-queues

Syntax max-queues *queues-per-line-card*;

Hierarchy Level [edit chassis fpc *slot-number*]

Release Information Statement introduced in Junos OS Release 13.2.
Support for MPC7E, MPC8E, and MPC9E from Junos OS Release 15.1F4.

Description Configure the maximum number of queues allowed per MPC1 Q, MPC2 Q, MPC2 EQ, MPC7E, MPC8E, and MPC9E line cards in an MX240, MX480, MX960, MX2010, or MX2020 router or per 2-port or 4-port 10-Gigabit Ethernet fixed MIC with XFP in an MX5, MX10, MX40, or MX80 modular chassis router.

Reducing the number of queues allowed in a hierarchical scheduling environment in turn reduces the degree of jitter in the queues.

CAUTION: **Committing a change to the maximum number of queues on a line card causes a reset of the line card.**

Options ***queues-per-line-card***—Maximum number of queues allowed for the line card. Only the following keywords are valid: **8k**, **16k**, **32k**, **64k**, **96k**, **128k**, **256k**, **512k**, **768k**, or **1M**.

- Built-in 10-Gigabit Ethernet MICs and MPC1 Q line cards support up to 128 K queues. You can use this statement to configure the single Packet Forwarding Engine to support a lower maximum number of queues.
- MPC2 Q and MPC2 EQ line cards support up to 256 K queues. You can use this statement to configure the two Packet Forwarding Engines to support a lower maximum number of queues.

If you configure a keyword for a value that exceeds the number of queues supported by the line card hardware, the system uses the maximum number of queues supported by the line card.

If the **max-queues** statement is *not* configured on MPC7E, MPC8E, and MPC9E, which is the default mode, the MPC starts with a message similar to the following:

FPC 0 supports only port based queuing. A license is required for per-VLAN and hierarchical features.

If the **max-queues** statement is configured on MPC7E, MPC8E, and MPC9E and the value is less than or equal to 32,000, the MPC starts with a message similar to the following:

Copyright © 2019, Juniper Networks, Inc.

FPC 0 supports port based queuing and is configured in 16384 queue mode. A limited per-VLAN queuing license is required for per VLAN and hierarchical queuing features.

If the **max-queues** statement is configured on MPC7E, MPC8E, and MPC9E and the value is greater than 32,000, the MPC starts with a message similar to the following:

FPC 0 supports port based queuing and is configured in 524288 queue mode. A full scale per-VLAN queuing license is required for per VLAN and hierarchical queuing features.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Jitter Reduction in Hierarchical CoS Queues on page 991
- *MPC1 Q*
- *MPC2 Q*
- *MPC2 EQ*
- *MIC/MPC Compatibility*
- *10-Gigabit Ethernet MICs with XFP*

Copyright © 2019, Juniper Networks, Inc.

max-queues-per-interface

Syntax max-queues-per-interface (8 | 4);

Hierarchy Level [edit chassis fpc *slot-number* pic *pic-number*],
[edit chassis lcc *number* fpc *slot-number* pic *pic-number*] (Routing Matrix)

Release Information Statement introduced before Junos OS Release 7.4.
Support for TX Matrix and TX Matrix Plus added in Junos OS Release 9.6.

Description On IQ, MPC, and DPC interfaces on M120, T320, T640, T1600, TX Matrix, and TX Matrix Plus routers, or on MIC or MPC interfaces on MX Series routers, set the number of egress queues per port to four or eight.

NOTE: If you include the **max-queues-per-interface 8** statement, the configuration at the **[edit class-of-service]** hierarchy level must also support eight queues per interface.

NOTE: When you include the **max-queues-per-interface** statement and commit the configuration, all physical interfaces on the PIC/MIC are deleted and readded. Also, the PIC/MIC is restarted automatically. You should change modes between four queues and eight queues only when there is no active traffic going to the PIC/MIC.

By default, IQ PICs on T Series and M320 routers are restricted to a maximum of four egress queues per interface. If you include the **max-queues-per-interface 4** statement, you can configure all four ports and configure up to four queues per port.

For Quad T3 and Quad E3 PICs and for 4-port OC3c/STM1 Type I and Type II PICs on M320 and T Series routers, when you include the **max-queues-per-interface 8** statement, you can configure up to eight queues on ports 0 and 2. After you commit the configuration, the PIC goes offline and comes back online with only ports 0 and 2 operational. No interfaces can be configured on ports 1 and 3.

NOTE: Starting from Junos OS Release 14.1R8, 14.2R6, 15.1F6, 15.1R3, 15.1R4, and 16.1R1, the restricted queue PICs without the **max-queues-per-interface** configuration boot up with a maximum of eight queues per port and two operational ports (port 0 and 2). PICs with restricted queues include Quad T3 PIC, Quad E3 PIC, 4-port SONET/SDH OC3c/STM1 PIC, and 4-Port OC3 and 1-port OC12 PICs with SFP.

Copyright © 2019, Juniper Networks, Inc.

On certain older MPCS (*MPC1 Q*, *MPC1E Q*, *MPC2 Q*, *MPC2E Q*), you can include the **max-queues-per-interface** statement to set the number of queues per logical interface to four or eight. Setting **max-queues-per-interface 4** sets the MPC to have four queues per logical interface and provides twice as many logical interfaces on the MPC as setting **max-queues-per-interface 8**.

NOTE: **For consistency, max-queues-per-interface should not be set on MPCs starting from Junos OS 14.1X51.**

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring the Junos OS to Support Eight Queues on IQ Interfaces for T Series and M320 Routers on page 744
- Configuring Up to 16 Custom Forwarding Classes on page 214
- Enabling Eight Queues on ATM Interfaces on page 842
- Configuring the Maximum Number of Queues for Trio MPC/MIC Interfaces on page 981
- *Example: Configuring CoS on SRX5000 Devices with an MPC*
- *Example: Enabling Eight-Queue Class of Service on Redundant Ethernet Interfaces on SRX Series Devices in a Chassis Cluster*

Copyright © 2019, Juniper Networks, Inc.

member-link-scheduler

Syntax
```
member-link-scheduler (replicate | scale);
```

Hierarchy Level
```
[edit class-of-service interfaces interface-name],
[edit logical-systems logical-system-name class-of-service interfaces interface-name],
[edit routing-instances routing-instance-name class-of-service interfaces interface-name]
```

Release Information Statement introduced in Junos OS Release 9.6.

Description Determines whether scheduler parameters for aggregated interface member links are applied in a replicated or scaled manner.

Default By default, scheduler parameters are scaled (in "equal division mode") among aggregated interface member links.

Options

replicate—Scheduler parameters are copied to each level of the aggregated interface member links.

scale—Scheduler parameters are scaled based on number of member links and applied each level of the aggregated interface member links.

Required Privilege Level

view-level—To view this statement in the configuration.
control-level—To add this statement to the configuration.

Related Documentation

- Configuring Hierarchical Schedulers for CoS on page 342

Copyright © 2019, Juniper Networks, Inc.

mode (Layer 2 Tunneling Protocol Shaping)

Syntax

```
mode traffic-manager-mode;
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number traffic-manager],
[edit chassis lcc number fpc slot-number pic pic-number traffic-manager]
[edit chassis fpc slot-number pic pic-number port port-number traffic-manager], (MX80 and MX104 routers only)
```

Release Information

Statement introduced in Junos OS Release 8.3.
Statement introduced at the [edit chassis fpc *slot-number* pic *pic-number* port *port-number* traffic-manager] hierarchy level on MX80 and MX104 routers in Junos OS Release 17.4R2.

Description

Enable shaping on an L2TP session.

Options

traffic-manager-mode—Configure CoS traffic manager mode of operation on this interface. This option has the following suboptions:

egress-only—Enable CoS queuing and scheduling on the egress side for the PIC that houses the interface. This is the default mode for an Enhanced Queuing (EQ) DPC on MX Series routers.

If ingress packet drops are observed at a high rate for an IQ2 or IQ2E PIC, configure the **traffic-manager** statement to work in the **egress-only** mode.

ingress-and-egress—Enable CoS queuing and scheduling on both the egress and ingress sides for the PIC. This is the default mode for IQ2 and IQ2E PICs on M Series and T Series routers.

Junos OS does not support **ingress-and-egress** mode on label-switched interfaces (LSIs) configured with VPLS.

For EQ DPCs, you must configure the **traffic-manager** statement with **ingress-and-egress** mode to enable ingress CoS on the EQ DPC. EQ DPCs have 250 ms of buffering, with only egress queuing (default mode). When **ingress-and-egress** is configured, the buffer is partitioned as 50 ms for the ingress direction and 200 ms for the egress direction.

session-shaping—(M10i and M120 routers only) Configure the IQ2 PIC mode for session-aware traffic shaping to enable L2TP session shaping.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Configuring Ingress Hierarchical CoS on MIC and MPC Interfaces on page 975
- Configuring CoS for L2TP Tunnels on ATM Interfaces on page 848
- egress-shaping-overhead on page 1098
- ingress-shaping-overhead on page 1170
- traffic-manager on page 1306

multilink-class

Syntax

```
multilink-class number;
```

Hierarchy Level

```
[edit class-of-service fragmentation-maps map-name forwarding-class class-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For link services IQ (**lsq**) interfaces only, map a forwarding class into a multiclass MLPPP (MCML).

The **multilink-class** statement and **no-fragmentation** statements are mutually exclusive.

Options

number—The multilink class assigned to this forwarding class.

Range: 0 through 7

Default: None

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Configuring CoS Fragmentation by Forwarding Class on LSQ Interfaces*
- *Configuring Multiclass MLPPP on LSQ Interfaces*
- Configuring Fragmentation by Forwarding Class on page 709
- *Junos OS Services Interfaces Library for Routing Devices*
- *multilink-max-classes*

Copyright © 2019, Juniper Networks, Inc.

next-hop (Class-Of-Service)

Syntax

```
next-hop [ next-hop-name ];
```

Hierarchy Level

```
[edit class-of-service forwarding-policy next-hop-map map-name forwarding-class
  class-name]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the next-hop name or address to which to map forwarded traffic.

Options ***next-hop-name***—Next-hop alias or IP address.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS-Based Forwarding on page 224

Copyright © 2019, Juniper Networks, Inc.

next-hop-map

Syntax

```
next-hop-map map-name {
  forwarding-class class-name {
    discard;
    lsp-next-hop [ lsp-regular-expression ];
    next-hop [next-hop-name];
    non-lsp-next-hop;
  }
  forwarding-class-default {
    discard;
    lsp-next-hop [ lsp-regular-expression ];
    next-hop [next-hop-name];
    non-lsp-next-hop;
  }
}
```

Hierarchy Level

[edit class-of-service forwarding-policy]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced for QFX10000 Series switches in Junos OS Release 17.1R1.

Description

Specify the map for CoS forwarding routes.

Options

map-name—Map that defines next-hop routes.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS-Based Forwarding on page 224

Copyright © 2019, Juniper Networks, Inc.

no-fragmentation

Syntax

```
no-fragmentation;
```

Hierarchy Level

[edit class-of-service fragmentation-maps forwarding-class *class-name*]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For link services IQ (**lsq**) interfaces only, set traffic on a particular forwarding class to be interleaved, rather than fragmented. This statement specifies that no extra fragmentation header is prepended to the packets received on this queue and that static-link load balancing is used to ensure in-order packet delivery.

Static-link load balancing is done based on packet payload. For IP version 4 (IPv4) and IP version 6 (IPv6) traffic, the link is chosen based on a hash computed from the source address, destination address, and protocol. If the IP payload is Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) traffic, the hash also includes source port and destination port. For MPLS traffic, the hash includes all MPLS labels and fields in the payload, whether the MPLS payload is IPv4 or IPv6.

Default

If you do not include this statement, the traffic in forwarding class ***class-name*** is fragmented.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Configuring CoS Fragmentation by Forwarding Class on LSQ Interfaces*

Copyright © 2019, Juniper Networks, Inc.

non-lsp-next-hop

Syntax non-lsp-next-hop;

Hierarchy Level [edit class-of-service forwarding-policy next-hop-map *map-name* forwarding-class *class-name*]

Release Information Statement introduced before Junos OS Release 9.0.

Description Use a non-LSP next hop for traffic sent to this forwarding class next-hop map of this forwarding policy.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring CoS-Based Forwarding on page 224

output-forwarding-class-map

Syntax output-forwarding-class-map *forwarding-class-map-name*;

Hierarchy Level [edit class-of-service forwarding-classes-interface-specific]

Release Information Statement introduced in Junos OS Release 9.6.

Description Apply a configured forwarding class map to a logical interface.

Options *forwarding-class-map-name*—Name of a forwarding class mapping configured at the **[edit class-of-service forwarding-classes-interface-specific]** hierarchy level.

Usage Guidelines "Classifying Packets by Egress Interface" on page 221

Required Privilege Level interface-control—To add this statement to the configuration.

Related Documentation
- forwarding-classes-interface-specific on page 1138

Copyright © 2019, Juniper Networks, Inc.

output-policer

Syntax

```
output-policer policer-name;
```

Hierarchy Level

[edit interfaces *interface-name* unit *logical-unit-number* layer2-policer],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* layer2-policer]

Release Information

Statement introduced in Junos OS Release 8.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Apply a single-rate two-color policer to the Layer 2 output traffic at the logical interface. The **output-policer** and **output-three-color** statements are mutually exclusive.

Options

policer-name—Name of the single-rate two-color policer that you define at the **[edit firewall]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Two-Color and Three-Color Policers at Layer 2*
- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 1033
- *Configuring a Gigabit Ethernet Policer*
- input-policer on page 1172
- input-three-color on page 1176
- layer2-policer on page 1187
- logical-interface-policer on page 1192
- output-three-color on page 1218

Copyright © 2019, Juniper Networks, Inc.

output-three-color

Syntax

```
output-three-color policer-name;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number layer2-policer]
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number
  layer2-policer]
```

Release Information

Statement introduced in Junos OS Release 8.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Apply a single-rate or two-rate three-color policer to the Layer 2 output traffic at the logical interface. The **output-three-color** and **output-policer** statements are mutually exclusive.

Options

policer-name—Name of the single-rate or two-rate three-color policer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Two-Color and Three-Color Policers at Layer 2*
- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 1033
- *Configuring a Gigabit Ethernet Policer*
- input-three-color on page 1176
- input-policer on page 1172
- layer2-policer on page 1187
- logical-interface-policer on page 1192
- output-policer on page 1217

Copyright © 2019, Juniper Networks, Inc.

output-traffic-control-profile

Syntax

output-traffic-control-profile *profile-name* shared-instance *instance-name*;

Hierarchy Level

[edit class-of-service interfaces *interface-name*],
[edit class-of-service interfaces *interface-name* unit *logical-unit-number*],
[edit class-of-service interfaces *interface-name* interface-set *interface-set-name*]

Release Information

Statement introduced in Junos OS Release 7.6.
interface-set option added for Enhanced Queuing DPCs on MX Series routers in Junos OS Release 8.5.
interface-set option added for MIC and MPC interfaces on MX Series routers in Junos OS Release 10.2.
Support on GRE tunnel interfaces configured on physical and logical interfaces on MICs or MPCs in MX Series routers added in Junos OS Release 13.3.

Description

Apply the specified CoS traffic control profile (traffic scheduling and shaping configuration objects) to the output traffic at the physical interface, logical interface, or interface set.

The statement is supported on the following interfaces:

- Channelized IQ PIC interfaces
- Gigabit Ethernet IQ, Gigabit Ethernet IQ2, and IQ2E PIC interfaces
- Link services IQ (LSQ) interfaces on Multiservices and Services PICs
- Enhanced Queuing DPC, MIC, and MPC interfaces on MX Series routers
- GRE tunnel interfaces configured on physical or logical interfaces hosted on MIC or MPC line cards in MX Series routers.

NOTE: Interface sets (sets of interfaces used to configure hierarchical CoS schedulers on supported Ethernet interfaces) are not supported on GRE tunnel interfaces.

The **shared-instance** statement is supported on Gigabit Ethernet IQ2 PICs only.

Options

profile-name—Name of the traffic-control profile to be applied to this interface.

shared-instance***instance-name***—Name of the shared scheduler and shaper to be applied to this interface.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Oversubscribing Interface Bandwidth on page 273
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- Configuring Hierarchical Schedulers for CoS on page 342 (Enhanced Queuing DPC, MIC, and MPC interfaces on MX Series routers)
- Configuring Interface Sets on page 264 (Enhanced Queuing DPC, MIC, and MPC interfaces on MX Series routers)
- output-traffic-control-profile-remaining on page 1221
- traffic-control-profiles on page 1305

Copyright © 2019, Juniper Networks, Inc.

output-traffic-control-profile-remaining

Syntax

output-traffic-control-profile-remaining *profile-name*;

Hierarchy Level

[edit class-of-service interfaces *interface-name*],
[edit class-of-service interfaces *interface-name* interface-set *interface-set-name*]

Release Information

Statement introduced in Junos OS Release 8.5.
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

Apply the specified traffic control profile (traffic scheduling and shaping configuration objects) to the remaining output traffic at the physical interface or interface set. The remaining traffic is transmitted by the default interface or interface set.

This statement is supported on the following interfaces:

- IQ2E PIC interfaces on M Series and T Series routers
- Enhanced Queuing DPC, MICs, and MPC interfaces on MX Series routers
- GRE tunnel interfaces configured on physical interfaces hosted on MIC or MPC line cards in MX Series routers.

NOTE: Interface sets (sets of interfaces used to configure hierarchical CoS schedulers on supported Ethernet interfaces) are not supported on GRE tunnel interfaces.

You can map the TCP to the interface or interface set by using the **output-traffic-control-profile-remaining** statement to explicitly configure the queues of the default interface or interface set scheduler that transmits the remaining traffic.

Options

profile-name—Name of the traffic-control profile for remaining traffic to be applied to this interface or interface set.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Hierarchical Schedulers for CoS on page 342
- Configuring Remaining Common Queues on MIC and MPC Interfaces on page 982
- output-traffic-control-profile on page 1219

Copyright © 2019, Juniper Networks, Inc.

overhead-accounting

Syntax

```
overhead-accounting {
  bytes bytes;
  cell-mode cell-mode-bytes cell-mode-bytes;
  frame-mode frame-mode-bytes frame-mode-bytes;
}
```

Hierarchy Level

[edit class-of-service traffic-control-profiles *profile-name*]

Release Information

Statement introduced in Junos OS Release 10.2.

Description

Configure the mode to shape downstream ATM traffic based on either frames or cells.

Default

The default is frame-mode.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Static Shaping Parameters to Account for Overhead in Downstream Traffic Rates on page 966
- *Configuring Dynamic Shaping Parameters to Account for Overhead in Downstream Traffic Rates*
- egress-shaping-overhead on page 1098

Copyright © 2019, Juniper Networks, Inc.

packet-timestamp

Syntax

```
packet-timestamp {
  enable;
}
```

Hierarchy Level

[edit chassis fpc *slot-number* traffic-manager]

Release Information

Statement introduced in Junos OS Release 16.1.

Description

On MX Series routers, enable timestamping of class-of-service (CoS) queues for a configured Flexible PIC Concentrator (FPC).

Options

enable—Display the last time a packet was enqueued as day, date, time, and year in the format *day-of-the-week month day-date hh:mm:ss yyyy*; for example, **Mon Jul 7 07:36:46 2014**. The Packet Forwarding Engine collects the timestamp for all inbound and outbound queue counters for all subscribers that are configured on the FPC and, when requested, also returns statistics corresponding to data traffic on the router.

If the timestamp is not enabled for the CoS queue, the **Last-packet enqueued** field is not displayed in statistics.

NOTE: When you commit the change after enabling or disabling the timestamp feature for a particular FPC *after* that is already up, the FPC is automatically restarted and the change takes effect when it is back up.

The Routing Engine generates a system log message in the following situations:

- When **packet-timestamp** is enabled and committed for an FPC slot, but the FPC inserted into that slot does not support timestamping.
- When **packet-timestamp** is enabled and committed for an FPC slot, but an FPC that does not support timestamping is later inserted into that slot.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Enabling a Timestamp for Ingress and Egress Queue Packets on page 594
- show interfaces queue on page 1421

Copyright © 2019, Juniper Networks, Inc.

peak-rate

Syntax peak-rate *rate*;

Hierarchy Level [edit class-of-service traffic-control-profiles *traffic-control-profile-name*]

Release Information Statement introduced in Junos OS Release 12.1.

Description (MX Series routers) Define ATM peak cell rate on ATM MICs in cells per second by entering a decimal number followed by the abbreviation c; where 1 cps = 384 bps.

Options *rate*—ATM peak rate in cells per second.

Range: 61 cps through 353,206 cps.

Range: (MX Series with MPCs and ATM MICs with SFP) 61 cps through 1,412,829 cps

NOTE: Beginning with Junos OS Release 14.2, to configure OC12 CBR bandwidth speed per virtual circuit (VC) on an ATM MIC with SFP (MIC-3D-8OC3-2OC12-ATM), specify up to 1,412,829 cps as the ATM peak cell rate.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *show class-of-service traffic-control-profile*

Copyright © 2019, Juniper Networks, Inc.

per-session-scheduler

Syntax

```
per-session-scheduler;
```

Hierarchy Level

[edit interfaces *interface-name* unit *logical-unit-number*]

Release Information

Statement introduced in Junos OS Release 8.3.

Description

Enable session-aware CoS shaping on this L2TP interface.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS for L2TP Tunnels on ATM Interfaces on page 848
- ingress-shaping-overhead on page 1170
- mode (Layer 2 Tunneling Protocol Shaping) on page 1211

Copyright © 2019, Juniper Networks, Inc.

per-unit-scheduler

Syntax per-unit-scheduler;

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 13.2 on 16x10GE MPC and MPC3E line cards.
Statement introduced in Junos OS Release 13.2 on PTX Series Packet Transport Routers.
Statement introduced in Junos OS Release 13.3 on MPC4E line cards.
Statement introduced in Junos OS Release 15.1 on MPC6E line cards.

Description For Channelized OC3 IQ, Channelized OC12 IQ, Channelized STM1 IQ, Channelized T3 IQ, Channelized E1 IQ, E3 IQ, link services IQ interfaces (lsq-), Gigabit Ethernet IQ, Gigabit Ethernet IQ2 and IQ2-E, and 10-, 40-, and 100-Gigabit Ethernet interfaces (including the 16x10GE MPC), enable the association of scheduler maps with logical interfaces.

CAUTION: Turning on per-unit scheduling causes the interface to reinitialize, which means all logical interfaces (units) on the interface are deleted and recreated.

NOTE: To enable per-unit scheduling on MX80 and MX104 routers, configure the **per-unit-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.

NOTE: Per-unit scheduling is not supported on T1 interfaces configured on the Channelized OC12 IQ PIC.

NOTE: On Gigabit Ethernet IQ2 and IQ2-E PICs without the **per-unit-scheduler** statement, the entire PIC supports 4071 VLANs and the user can configure all the VLANs on the same port.

On Gigabit Ethernet IQ2 and IQ2-E PICs with the **per-unit-scheduler** statement, the entire PIC supports 1024 – 2 * number of ports (1024 minus two times the number of ports), because each port is allocated two default schedulers.

Copyright © 2019, Juniper Networks, Inc.

When including the **per-unit-scheduler** statement, you must also include the **vlan-tagging** statement or the **flexible-vlan-tagging** statement (to apply scheduling to VLANs) or the **encapsulation frame-relay** statement (to apply scheduling to DLCIs) at the **[edit interfaces *interface-name*]** hierarchy level.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 300
- vlan-tagging on page 1324
- flexible-vlan-tagging on page 1127
- Example: Applying Scheduling and Shaping to VLANs on page 312
- Configuring Virtual LAN Queuing and Shaping on PTX Series Routers on page 631

plp-to-clp

Syntax
```
plp-to-clp;
```

Hierarchy Level
```
[edit interfaces at-fpc/pic/port atm-options],
[edit interfaces at-fpc/pic/port unit logical-unit-number],
[edit logical-systems logical-system-name interfaces at--fpc/pic/port unit
  logical-unit-number]
```

Release Information
Statement introduced before Junos OS Release 7.4.

Description
For ATM2 IQ interfaces only, enable the PLP setting to be copied to the cell-loss priority (CLP) bit.

Default
If you omit this statement, the Junos OS does not copy the PLP setting to the CLP bit.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- *Enabling the PLP Setting to Be Copied to the CLP Bit*
- Copying the Packet Loss Priority to the CLP Bit on ATM Interfaces on page 848

Copyright © 2019, Juniper Networks, Inc.

policer (Configuring)

Syntax

```
policer policer-name {
  filter-specific;
  if-exceeding {
    bandwidth-limit bps;
    bandwidth-percent number;
    burst-size-limit bytes;
  }
  logical-bandwidth-policer;
  logical-interface-policer;
  physical-interface-policer;
  shared-bandwidth-policer;
  then {
    policer-action;
  }
}
```

Hierarchy Level

```
[edit dynamic-profiles profile-name firewall],
[edit firewall],
[edit logical-systems logical-system-name firewall]
```

Release Information

Statement introduced before Junos OS Release 7.4.
The **out-of-profile** policer action added in Junos OS Release 8.1.
The **logical-bandwidth-policer** statement added in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
The **physical-interface-policer** statement introduced in Junos OS Release 9.6.
The **shared-bandwidth-policer** statement added in Junos OS Release 11.2.
Support at the **[edit dynamic-profiles ... firewall]** hierarchy level introduced in Junos OS Release 11.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Configure policer rate limits and actions. When included at the **[edit firewall]** hierarchy level, the **policer** statement creates a template, and you do not have to configure a policer individually for every firewall filter or interface. To activate a policer, you must include the **policer-action** modifier in the **then** statement in a firewall filter term or on an interface.

You can configure the policer in static firewall filters or dynamic firewall filters in a dynamic client profile or a dynamic service profile.

Options

policer-action—One or more actions to take:

- **discard**—Discard traffic that exceeds the rate limits.
- **forwarding-class *class-name***—Specify the particular forwarding class.
- **loss-priority**—Set the packet loss priority (PLP) to **low**, **medium-low**, **medium-high**, or **high**.

Copyright © 2019, Juniper Networks, Inc.

policer-name—Name that identifies the policer. The name can contain letters, numbers, and hyphens (-), and can be up to 255 characters long. To include spaces in the name, enclose it in quotation marks (" "). Policer names cannot begin with an underscore in the form **__.***.

then—Actions to take on matching packets.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- *Bandwidth Policer Overview*
- *Configuring Firewall Filters and Policers for VPLS*
- Configuring Multifield Classifiers on page 98
- *Logical Interface (Aggregate) Policer Overview*
- *Physical Interface Policer Overview*
- *Single-Rate Two-Color Policer Overview*
- Using Multifield Classifiers to Set Packet Loss Priority on page 101
- **filter (Configuring) on page 1124**
- **priority (Schedulers) on page 1235**

Copyright © 2019, Juniper Networks, Inc.

policing-action

Syntax

```
policing-action (discard | discard-tag | count);
```

Hierarchy Level

```
[edit firewall atm-policer policer-name]
```

Release Information

Statement introduced in Junos OS Release 12.1.

Description

Configure the policing action to be taken when the traffic exceeds the limits set for the policer.

This action is associated with the ATM policer only if policing is enabled on the ATM interface.

Options

discard—Discard traffic at ingress that exceeds the rate limit. These packets are discarded but not counted.

discard-tag—Discard and tag packets at ingress that exceed the rate limit. These packets are discarded and counted.

count—Count all the packets that are received at ingress.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- *show class-of-service traffic-control-profile*

Copyright © 2019, Juniper Networks, Inc.

policy-map

Syntax

```
policy-map policy-map-name {
  inet-precedence proto-ip code-point [alias | bits];
  inet-precedence proto-mpls code-point [alias | bits];
  dscp proto-ip code-point [alias | bits];
  dscp proto-mpls code-point [alias | bits];
  dscp-ipv6 proto-ip code-point [alias | bits];
  dscp-ipv6 proto-mpls code-point [alias | bits];
  exp all-label code-point [alias | bits];
  exp outer-label code-point [alias | bits];
  ieee-802.1 outer code-point [alias | bits];
  ieee-802.1 outer-and-inner code-point [alias | bits];
  ieee-802.1ad outer code-point [alias | bits];
  ieee-802.1ad outer-and-inner code-point [alias | bits];
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 16.1R1.

Description

Define the packet- marking scheme (rewrite rules) for a customer. Traditionally, packet marking in the Junos OS uses the forwarding class and loss priority determined from a behavior aggregate (BA) classifier or multifield classifier. This approach does not allow rewrite rules to be directly assigned for each customer because of the limited number of combinations of forwarding class and loss priority. **policy map** enables you to define rewrite rules on a per-customer basis.

NOTE: Creating a policy map requires **enhanced-ip**, **enhanced-ethernet**, or **enhanced-mode** to be configured under **[edit chassis network-services]**.

NOTE: Policy maps have the following configuration restrictions:

- When configuring both **proto-ip** and **proto-mpls** options for **inet-precedence**, **dscp**, or **dscp-ipv6**, you must configure both options with the same code point or code point alias.
- You cannot configure **inet-precedence** and **dscp** in the same policy map.
- You cannot configure **ieee-802.1** and **ieee-802.1ad** in the same policy map.
- You cannot configure both **outer** and **outer-and-inner** options for **ieee-802.1** and **ieee-802.1ad** code points in the same policy map.

Copyright © 2019, Juniper Networks, Inc.

You assign the policy map to a customer through a firewall action on an ingress or egress firewall filter (where the match conditions identify the customer). Alternatively, you can also assign a policy map to an ingress interface or a routing instance. A policy map is executed on a packet just before it is queued, so it overrides any other packet-marking scheme that was previously applied to the packet.

Options

alias—The predefined CoS code point alias on the device.

bits—The bit pattern for the CoS code point.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- interfaces (CoS) on page 1181
- routing-instances (CoS) on page 1252

priority (ATM2 IQ Schedulers)

Syntax

```
priority (high | low);
```

Hierarchy Level

```
[edit interfaces at-fpc/pic/port atm-options scheduler-maps map-name forwarding-class
  class-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For ATM2 IQ interfaces only, assign queuing priority to a forwarding class.

Options

low—Forwarding class has low priority.

high—Forwarding class has high priority.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Applying Scheduler Maps to ATM Interfaces on page 853

Copyright © 2019, Juniper Networks, Inc.

priority (Fabric Priority)

Syntax

```
priority (high | low);
```

Hierarchy Level

[edit class-of-service forwarding-classes class *class-name* queue-num *queue-number*],
[edit class-of-service forwarding-classes queue *queue-number class-name*]

Release Information

Statement introduced before Junos OS Release 7.4.
[edit class-of-service forwarding-classes class *class-name* queue-num *queue-number*] hierarchy level added in Junos OS Release 8.1.

Description

For M320 routers, MX Series routers, T Series routers and EX Series switches only, specify a fabric priority value.

The two hierarchy levels are mutually exclusive. To configure up to eight forwarding classes with one-to-one mapping between forwarding classes and output queues, include this statement at the **[edit class-of-service forwarding-classes queue *queue-number class-name*]** hierarchy level. To configure up to 16 forwarding classes with multiple forwarding classes mapped to single queues, include this statement at the **[edit class-of-service forwarding-classes class *class-name* queue-num *queue-number*]** hierarchy level.

Options

low—Forwarding class's fabric queuing has low priority.

high—Forwarding class's fabric queuing has high priority.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Forwarding Classes and Fabric Priority Queues on page 234
- Configuring Up to 16 Custom Forwarding Classes on page 214

Copyright © 2019, Juniper Networks, Inc.

priority (Fabric Queues, Schedulers)

Syntax priority (high | low)scheduler *scheduler-name*;

Hierarchy Level [edit class-of-service fabric scheduler-map]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS 11.4 for EX Series switches.

Description Define Fabric traffic priority. For M320, MX Series, T Series routers and EX Series switches only, specify the fabric priority with which a scheduler is associated.

For a scheduler that you associate with a fabric priority, you cannot include the **buffer-size**, **transmit-rate**, or **priority** statements at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level.

On EX Series switches, this statement is supported only on EX8200 standalone switches and EX8200 Virtual Chassis.

Options **high**—Scheduler has high priority.

low—Scheduler has low priority.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- See Associating Schedulers with Fabric Priorities on page 331.

Copyright © 2019, Juniper Networks, Inc.

priority (Schedulers)

Syntax priority *priority-level*;

Hierarchy Level [edit class-of-service schedulers *scheduler-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.

Description Specify the packet-scheduling priority value.

Options ***priority-level*** can be one of the following:

- **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.

NOTE: The **strict-high** priority level is the only priority level supported on ACX Series Routers. However, multiple strict-high priority queues can be configured per interface on ACX Series Routers.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Schedulers for Priority Scheduling on page 330

Copyright © 2019, Juniper Networks, Inc.

protocol (Host Outbound Traffic)

Syntax

```
protocol {
  isis-over-gre {
    dscp-code-point dscp-code-point;
  }
}
```

Hierarchy Level

[edit class-of-service host-outbound-traffic],
[edit dynamic-profiles *name* class-of-service host-outbound-traffic]

Release Information

Statement introduced in Junos OS Release 17.3.

Description

Set the DSCP code point for IS-IS traffic originating on the host and being sent over a GRE tunnel. By default, this DSCP code point is **000000**, which is normally mapped to **best-effort** forwarding class and can lead to packet drops during congestion. Set this DSCP code point to enable traffic prioritization for IS-IS traffic originating on a host and being sent over a GRE tunnel.

Required Privilege Level

interface

Copyright © 2019, Juniper Networks, Inc.

protocol (Rewrite Rules)

Syntax

```
protocol protocol-types;
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules exp
  rewrite-name],
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules dscp
  rewrite-name],
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules
  dscp-ipv6 rewrite-name],
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules
  inet-precedence rewrite-name]
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules
  inet6-precedence rewrite-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Option for **dscp** and **inet-prec** introduced in Junos OS Release 8.4.
Option for **dscp-ipv6** introduced in Junos OS Release 10.4R2.

Description

Apply a rewrite rule to MPLS packets only, and write the CoS value to MPLS headers only; or apply a rewrite rule to MPLS and IPv4 or IPv6 packets, and write the CoS value to MPLS and IPv4 or IPv6 headers.

Options

protocol-types can be one of the following:

- **mpls**—Apply a rewrite rule to MPLS packets and write the CoS value to MPLS headers.
- **mpls-inet-both**—Apply a rewrite rule to VPN MPLS packets with IPv4 payloads. On M120, M320, MX Series, and T Series routers (except T4000 routers), and EX Series switches, write the CoS value to the MPLS and IPv4 headers. On M Series routers, initialize all ingress MPLS LSP packets with IPv4 payloads with 000 code points for the MPLS EXP value, and the configured rewrite code point for IP precedence.
- **mpls-inet-both-non-vpn**—Apply a rewrite rule to non-VPN MPLS packets with IPv4 payloads. On M120, M320, MX Series, T Series routers, and EX Series switches write the CoS value to the MPLS and IPv4 headers. On M Series routers, initialize all ingress MPLS LSP packets with IPv4 payloads with 000 code points for the MPLS EXP value, and the configured rewrite code point for IP precedence.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Rewriting MPLS and IPv4 Packet Headers on page 403

Copyright © 2019, Juniper Networks, Inc.

protocol (Schedulers)

Syntax

```
protocol (any | non-tcp | tcp);
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name drop-profile-map]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.

Description

Specify the protocol type for the specified scheduler.

Options

any—Accept any protocol type.

non-tcp—(ACX Series Routers, M Series and T Series (except T4000) routers only) Accept any protocol type other than TCP/IP.

NOTE: On ACX Series Routers, when you configure the non-tcp option, only the any option is supported for loss-priority.

tcp—(ACX Series Routers, M Series and T Series (except T4000) routers only) Accept TCP/IP protocol type.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Schedulers on page 258

Copyright © 2019, Juniper Networks, Inc.

queue (Global Queues)

Syntax

```
queue queue-number class-name;
```

Hierarchy Level

```
[edit class-of-service forwarding-classes]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.

Description

Specify the output transmission queue to which to map all input from an associated forwarding class.

On M120, M320, MX Series, T Series routers and on EX Series switches, this statement enables you to configure up to eight forwarding classes with one-to-one mapping to output queues. If you want to configure up to 16 forwarding classes with multiple forwarding classes mapped to single output queues, include the **class** statement instead of the **queue** statement at the **[edit class-of-service forwarding-classes]** hierarchy level.

Options

class-name—Name of forwarding class.

queue-number—Output queue number.

Range: 0 through 7. For M Series routers and some T Series router PICs, 0 through 3.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring a Custom Forwarding Class for Each Queue on page 212
- class (Forwarding Classes) on page 1064

Copyright © 2019, Juniper Networks, Inc.

queue (Restricted Queues)

Syntax queue *queue-number*;

Hierarchy Level [edit class-of-service restricted-queues forwarding-class *class-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For M320, MX Series, T Series routers and EX Series switches only, map forwarding classes to restricted queues.

Options ***queue-number***—Output queue number.

Range: 0 through 3.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

queue-depth

Syntax queue-depth *cells*;

Hierarchy Level [edit interfaces *interface-name* atm-options linear-red-profiles *profile-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM2 IQ interfaces only, define maximum queue depth in the CoS VC drop profile. Packets are always dropped beyond the defined maximum. This statement is mandatory; there is no default configuration.

Default Buffer usage is unregulated.

Options ***cells***—Maximum number of cells the queue can contain.
Range: 1 through 64,000 cells

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *ATM2 IQ VC Tunnel CoS Components Overview*
- Configuring Linear RED Profiles on ATM Interfaces on page 859
- *high-plp-threshold*
- low-plp-threshold on page 1203

Copyright © 2019, Juniper Networks, Inc.

queue-threshold

Syntax

```
queue-threshold {
  fabric-queue {
    priority priority {
      threshold threshold-percentage;
    }
  }
  wan-queue {
    priority priority {
      threshold threshold-percentage;
    }
  }
}
```

Hierarchy Level [edit chassis fpc *slot-number* traffic-manager]

Release Information Statement introduced in Junos OS Release 15.1.

Description Enable monitoring of fabric and WAN queues.

When the **fabric-queue** statement is configured, an SNMP trap is generated when the fabric queue depth exceeds the configured threshold value.

When **wan-queue** is configured, an SNMP trap is generated whenever the WAN queue depth exceeds the configured threshold value.

By default, monitoring of fabric queue and WAN queue is disabled.

Options

fabric-queue—Configure threshold values to monitor fabric queue depth. An SNMP trap is generated when fabric queue depth exceeds the configured threshold value. When this option is configured, you can use the show class-of-service fabric statistics **detail** command or the **show pfe statistics traffic detail** command to display detailed statistics at the level of the Packet Forwarding Engine.

- **priority *priority***—Specify the queue priority for fabric queues. The options are **high** and **low**. Traffic with **low** priority is sent after any queue that has a **high** priority. The priorities map to numeric priorities in the underlying hardware.
- **threshold *threshold-percentage***—Configured threshold value for the priority level for fabric queues. An SNMP trap is generated when the actual queue utilization percentage exceeds the configured threshold value.

wan-queue—Configure threshold values to monitor the WAN queue depth. An SNMP trap is generated when the WAN queue depth exceeds the configured value.

- **priority *priority***—Specify the queue priority for WAN queues. The priority level can be **high**, **medium-high**, **medium-low**, and **low**.

Copyright © 2019, Juniper Networks, Inc.

- **threshold *threshold-percentage***—Configured threshold value for the priority level for WAN queues. An SNMP trap is generated when the actual queue utilization percentage exceeds the configured threshold value.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- traffic-manager on page 1306

Copyright © 2019, Juniper Networks, Inc.

red-buffer-occupancy

Syntax

```
red-buffer-occupancy {
  weighted-averaged [ instant-usage-weight-exponent exponent-value ];
}
```

Hierarchy Level

```
[edit chassis fpc slot-number pic pic-number ],
[edit chassis lcc number fpc slot-number pic pic-number]
```

Release Information Statement introduced in Junos OS Release 8.3.

Description Configure the IQ PIC to base random early detection (RED) queue management on a *simple moving average* buffer occupancy calculation. If you do not include this statement, the IQ PIC bases RED on an *instantaneous* buffer occupancy value. As an option, you can specify that the IQ PIC bases RED on a *weighted moving average* of buffer occupancy values.

If you configure this feature on a channelized OC12 intelligent queuing (IQ) PIC, the PIC reboots.

Options

weighted-averaged—Configure the IQ PIC to base RED processing on a simple moving average of instantaneous buffer occupancy values instead of an instantaneous buffer occupancy.

instant-usage-weight-exponent *exponent-value*—(Optional) Specify the integer to be used as the negative exponent of 2 to express a weight value. The PIC performs weighted RED (WRED) by based on a calculation of average buffer occupancy that applies the specified weight value to the instantaneous buffer occupancy and then factors the weighted value into the calculation of average buffer occupancy. Valid exponent range is from 1 through 31 (weight values from 2^{-1} through 2^{-31}). If you do not specify this option, the default exponent value is 0, which results in a weight value of 2^{0} = 1. With a weight value of 1, the calculation of weighted average buffer occupancy yields the same value as the simple average buffer occupancy.

NOTE: You can specify an exponent value greater than 31, and the value displays in the output of show commands. However, the PIC replaces the out-of-range value with the *operational* value of 31, which results in a weight value of $2^{-31} = 1 / 2^{31}$ = 0.00000000046566128730773925781 25.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 381
- Example: Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 383

reflexive | revert | reverse

Syntax

```
reflexive; | revert; | reverse {
  application-profile profile-name;
  dscp (alias | bits);
  forwarding-class class-name;
  syslog;
}
```

Hierarchy Level

[edit services cos rule *rule-name* term *term-name* then]

Release Information

Statement introduced in Junos OS Release 8.1.
revert option introduced in Junos OS Release 16.1R5 and 17.4R1.

Description

reflexive—Applies the CoS rule actions to flows in the reverse direction as well as to flows in the matching direction.

revert—Stores the DSCP and forwarding class of a packet that is received in the match direction of the rule and then applies that DSCP and forwarding class to packets that are received in the reverse direction of the same session.

reverse—Allows you to define CoS behavior for flows in the reverse direction.

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS Rules on Services PICs on page 699
- *Configuring CoS Rules*

Copyright © 2019, Juniper Networks, Inc.

restricted-queues

Syntax

```
restricted-queues {
  forwarding-class class-name queue queue-number;
}
```

Hierarchy Level

[edit class-of-service]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For M320, MX Series, T Series routers and EX Series switches only, map forwarding classes to restricted queues.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

rewrite-rules (CoS Host Outbound Traffic)

Syntax

```
rewrite-rules;
```

Hierarchy Level

```
[edit class-of-service host-outbound-traffic ieee-802.1]
```

Release Information Statement introduced in Junos OS Release 12.3.

Description Apply the IEEE 802.1p rewrite rules associated with the egress logical interface to the IEEE 802.1p PCP field for all host outbound traffic on that interface.

NOTE: Enabling IEEE 802.1p rewrite rules for host outbound traffic on a DPC without creating any corresponding IEEE 802.1p rewrite rules on a logical interface on the DPC causes the IEEE 802.1p code point to be automatically set to 000 for all host generated traffic that exits that logical interface.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 597
- Rewriting Packet Headers to Ensure Forwarding Behavior on page 388
- Configuring Rewrite Rules on page 391

Copyright © 2019, Juniper Networks, Inc.

rewrite-rules (Definition)

Syntax

```
rewrite-rules {
  type rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
      loss-priority level code-point [ aliases ] [ bit-patterns ];
    }
  }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced before Junos OS Release 7.4.
ieee-802.1ad option introduced in Junos OS Release 9.2.

Description

Specify a rewrite-rules mapping for the traffic that passes through all queues on the interface.

Options

rewrite-name—Name of a **rewrite-rules** mapping.

type—Traffic type.

Values: **dscp**, **dscp-ipv6**, **exp**, **ieee-802.1**, **ieee-802.1ad**, **inet-precedence**, **inet6-precedence**

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Rewrite Rules on page 391

Copyright © 2019, Juniper Networks, Inc.

rewrite-rules (Interfaces)

Syntax

```
rewrite-rules {
  dscp (rewrite-name | default) protocol (inet-both | inet-outer | mpls);
  dscp-ipv6 (rewrite-name | default) protocol mpls;
  exp (rewrite-name | default) protocol protocol-types;
  exp-push-push-push default;
  exp-swap-push-push default;
  ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
  ieee-802.1ad (rewrite-name | default) vlan-tag (outer | outer-and-inner);
  inet-precedence (rewrite-name | default) protocol (inet-both | inet-outer | mpls);
}
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name],
[edit class-of-service interfaces interface-name unit logical-unit-number]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description Associate a rewrite-rules configuration or default mapping with a specific interface.

The **[edit class-of-service interfaces *interface-name*]** hierarchy level is not supported on M Series routers.

The **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level is not supported on ACX Series routers.

Integrated Bridging and Routing (IRB) interfaces are used to tie together Layer 2 switched and Layer 3 routed domains on MX routers. MX routers support classifiers and rewrite rules on the IRB interface at the **[edit class-of-service interfaces irb unit logical-unit-number]** level of the hierarchy. All types of classifiers and rewrite rules are allowed, including IEEE 802.1p.

NOTE: The IRB classifiers and rewrite rules are used only for *routed* packets; in other words, it is for traffic that originated in the Layer 2 domain and is then routed through IRB into the Layer 3 domain, or vice versa. Only IEEE classifiers and IEEE rewrite rules are allowed for pure Layer 2 interfaces within a bridge domain.

On an MX Series router and on an EX Series switch, **exp-push-push-push**, **exp-swap-push-push**, and **frame-relay-de** are not supported on an integrated routing and bridging (IRB) interface.

On an ACX Series router, only the outer tag is supported for **dscp**, **inet-precedence**, and **ieee802.1**.

Copyright © 2019, Juniper Networks, Inc.

On M Series routers only, if you include the **control-word** statement at the **[edit protocols l2circuit neighbor *address* interface *interface-name*]** hierarchy level, the software cannot rewrite MPLS EXP bits.

For IQ PICs, you can configure only one IEEE 802.1 rewrite rule on a physical port. All logical ports (units) on that physical port should apply the same IEEE 802.1 rewrite rule.

On M320 and T Series routers (except for T4000 routers with Type 5 FPCs), for a single interface, you cannot enable a rewrite rule on a subset of forwarding classes. You must assign a rewrite rule to either none of the forwarding classes or all of the forwarding classes. When you assign a rewrite rule to a subset of forwarding classes, the commit does not fail, and the subset of forwarding classes works as expected. However, the forwarding classes to which the rewrite rule is not assigned are rewritten to all zeros.

For example, if you configure a Differentiated Services code point (DSCP) rewrite rule, the bits in the forwarding classes to which you do not assign the rewrite rule are rewritten to 000000. If you configure an IP precedence rewrite rule, the bits in the forwarding classes to which you do not assign the rewrite rule are rewritten to 000.

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. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Rewrite Rules on page 391
- **rewrite-rules (Definition) on page 1248**
- Applying Rewrite Rules to Output Logical Interfaces on page 401

Copyright © 2019, Juniper Networks, Inc.

rewrite-rules (Physical Interfaces)

Syntax

```
rewrite-rules {
  dscp (rewrite-name | default);
  ieee-802.1 (rewrite-name | default);
  inet-precedence (rewrite-name | default);
}
```

Hierarchy Level

[edit class-of-service interfaces *interface-name*]

Release Information

Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.

Description

Associate a rewrite-rules configuration or default mapping with a specific interface.

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. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

routing-instances (CoS)

Syntax

```
routing-instances routing-instance-name {
  classifiers {
    dscp (classifier-name | default);
    dscp-ipv6 (classifier-name | default);
    exp (classifier-name | default);
    ieee-802.1 (classifier-name | default);
    no-default;
  }
  policy-map policy-map-name{
  }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced before Junos OS Release 7.4.
no-default and **policy-map** options added for MX Series devices only in Junos OS Release 16.1.

Description

For routing instances with VRF table labels enabled, apply a DSCP, MPLS EXP, or IEEE 802.1 classifier to the routing instance. You can apply the default classifier or one that is previously defined.

Apply the **no-default** option to disable the application of any default classifier to the routing instance.

You can also apply a policy map that defines the rewrite rules for a routing instance.

Default

If you do not include this statement with the **classifiers** option, the default MPLS EXP classifier is applied to the routing instance. When no DSCP classifier is configured, the default MPLS EXP classifier is applied.

Options

routing-instance-name—Name of the routing instance.

classifier-name—Name of the behavior aggregate MPLS EXP classifier or DSCP classifier.

policy-map-name—Name of the policy map that defines the rewrite rules for a routing instance.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Documentation

- Configuring a Custom Forwarding Class for Each Queue on page 212
- *Applying Custom MPLS EXP Classifiers to Routing Instances in Layer 3 VPNs*

Copyright © 2019, Juniper Networks, Inc.

rtvbr

Syntax

```
rtvbr peak rate sustained rate burst length;
```

Hierarchy Level

```
[edit interfaces interface-name atm-options vpi vpi-identifier shaping ],
[edit interfaces interface-name unit logical-unit-number address address family family
multipoint-destination address shaping ],
[edit interfaces interface-name unit logical-unit-number shaping ],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number
  shaping ],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number
  address address family family multipoint-destination address shaping ]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM2 IQ PICs only, define the real-time variable bandwidth utilization in the traffic-shaping profile.

When you configure the real-time bandwidth utilization, you must specify all three options (**burst**, **peak**, and **sustained**). You can specify the rate 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). You can also specify the rate in cells per second by entering a decimal number followed by the abbreviation **c**; values expressed in cells per second are converted to bits per second using the formula 1 cps = 384 bps.

Default If the **rtvbr** statement is not included, bandwidth utilization is unlimited.

Options

burst *length*—Burst length, in cells. If you set the length to 1, the peak traffic rate is used.

Range: 1 through 4000 cells

peak *rate*—Peak rate, in bits per second or cells per second.

Range: For ATM2 IQ OC3 and OC12 interfaces, 33 Kbps through 542,526,792 bps. For ATM2 IQ OC48 interfaces, 33 Kbps through 2,170,107,168 bps. For ATM2 IQ DS3 and E3 interfaces, 33 Kbps through the maximum rate, which depends on the ATM encapsulation and framing you configure..

sustained *rate*—Sustained rate, in bps or cps.

Range: For ATM2 IQ OC3 and OC12 interfaces, 33 Kbps through 542,526,792 bps. For ATM2 IQ OC48 interfaces, 33 Kbps through 2,170,107,168 bps. For ATM2 IQ DS3 and E3 interfaces, from 33 Kbps through the maximum rate, which depends on the ATM encapsulation and framing you configure.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- *Configuring ATM CBR*
- *Configuring ATM2 IQ Real-Time VBR*
- Applying Scheduler Maps to Logical ATM Interfaces on page 858
- **cbr on page 1060**
- **vbr on page 1319**

Copyright © 2019, Juniper Networks, Inc.

rule (Services CoS)

Syntax

```
rule rule-name {
  match-direction (input | output | input-output);
  term term-name {
    from {
      application-sets set-name;
      applications [ application-names ];
      destination-address address;
      destination-prefix-list list-name <except>;
      source-address address;
      source-prefix-list list-name <except>;
    }
    then {
      application-profile profile-name;
      dscp (alias | bits);
      forwarding-class class-name;
      syslog;
      reflexive; | revert; | reverse {
        application-profile profile-name;
        dscp (alias | bits);
        forwarding-class class-name;
        syslog;
      }
    }
  }
}
```

Hierarchy Level

```
[edit services cos],
[edit services cos rule-set rule-set-name]
```

Release Information

Statement introduced in Junos OS Release 8.1.

Description

Specify the rule the router uses when applying this service.

Options

rule-name—Identifier for the collection of terms that constitute this rule.

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Configuring CoS Rules*
- Configuring CoS Rules on Services PICs on page 699

Copyright © 2019, Juniper Networks, Inc.

rule-set (Services CoS)

Syntax

```
rule-set rule-set-name {
   [ rule rule-name ];
}
```

Hierarchy Level

[edit services cos]

Release Information

Statement introduced in Junos OS Release 8.1.

Description

Specify the rule set the router uses when applying this service.

Options

rule-set-name—Identifier for the collection of rules that constitute this rule set.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS Rule Sets on Services PICs on page 705

scheduler (Fabric Queues)

Syntax

```
scheduler scheduler-name;
```

Hierarchy Level

[edit class-of-service fabric scheduler-map priority (high | low)]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS Release 11.4 for EX Series switches.

Description

Define scheduler name. For M320, MX Series, T Series routers and for EX Series switches only, specify a scheduler to associate with a fabric queue. For fabric CoS configuration, schedulers are restricted to transmit rates and drop profiles.

Options

scheduler-name—Name of the scheduler configuration block.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- See Associating Schedulers with Fabric Priorities on page 331.

Copyright © 2019, Juniper Networks, Inc.

scheduler (Scheduler Map)

Syntax
```
scheduler scheduler-name;
```

Hierarchy Level
```
[edit class-of-service scheduler-maps map-name]
```

Release Information
Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.

Description
Associate a scheduler with a scheduler map.

Options
scheduler-name—Name of the scheduler configuration block.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Schedulers on page 258

scheduler-map (Fabric Queues)

Syntax
```
scheduler-map priority (high | low) scheduler scheduler-name;
```

Hierarchy Level
```
[edit class-of-service fabric]
```

Release Information
Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS Release 11.4 for EX Series switches.

Description
Mapping of fabric traffic to packet schedulers. For M320, MX Series, T Series routers, and for EX Series switches only, associate a scheduler with a fabric priority.

On EX Series switches, this statement is supported only on EX8200 standalone switches and EX8200 Virtual Chassis.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- See Associating Schedulers with Fabric Priorities on page 331.

Copyright © 2019, Juniper Networks, Inc.

scheduler-map (Interfaces and Traffic-Control Profiles)

Syntax

```
scheduler-map map-name;
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name],
[edit class-of-service interfaces interface-name unit logical-unit-number],
[edit class-of-service traffic-control-profiles]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description For Gigabit Ethernet IQ, Channelized IQ PICs, and FRF.15 and FRF.16 LSQ interfaces only, associate a scheduler map name with an interface or with a traffic-control profile.

For channelized OC12 intelligent queuing (IQ), channelized T3 IQ, channelized E1 IQ, and Gigabit Ethernet IQ interfaces only, you can associate a scheduler map name with a logical interface.

Options *map-name*—Name of the scheduler map.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Schedulers on page 258
- Oversubscribing Interface Bandwidth on page 273
- output-traffic-control-profile on page 1219

Copyright © 2019, Juniper Networks, Inc.

scheduler-map-chassis

Syntax scheduler-map-chassis (derived | *map-name);*

Hierarchy Level [edit class-of-service interfaces *interface-type-fpc/pic/**]

Release Information Statement introduced before Junos OS Release 7.4.

Description For IQ and IQ2 interfaces, as well as on the 10x10GE MIC with SFP+, assign a custom scheduler to the packet forwarding component queues that control the aggregated traffic transmitted into the entire PIC.

Default On Intelligent Queuing (IQ) and Intelligent Queuing 2 (IQ2) interfaces, as well as on the 10x10GE MIC with SFP+, the traffic that is fed from the packet forwarding components into the PIC uses low packet loss priority (PLP) by default and is distributed evenly across the four chassis queues (not PIC queues), regardless of the scheduling configuration for each logical interface. This default behavior can cause traffic congestion.

Options

derived—Sets the chassis queues to derive their scheduling configuration from the associated logical interface scheduling configuration.

map-name—Name of the scheduler map configured at the **[edit class-of-service scheduler-maps]** hierarchy level.

BEST PRACTICE: It is always preferred that you apply a custom scheduler map instead of using the derived option. We strongly recommend that you do *not* use the derived option if the PIC is oversubscribed.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Applying Scheduler Maps to Chassis-Level Queues on page 778
- scheduler-map (Fabric Queues) on page 1257

Copyright © 2019, Juniper Networks, Inc.

scheduler-maps (For ATM2 IQ Interfaces)

Syntax

```
scheduler-maps map-name {
  forwarding-class (class-name | assured-forwarding | best-effort | expedited-forwarding
    | network-control);
  vc-cos-mode (alternate | strict);
}
```

Hierarchy Level

```
[edit at-fpc/pic/port interface-name atm-options]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For ATM2 IQ interfaces only, define CoS parameters assigned to forwarding classes.

Options

map-name—Name of the scheduler map.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *ATM2 IQ VC Tunnel CoS Components Overview*
- Applying Scheduler Maps to ATM Interfaces on page 853
- atm-scheduler-map on page 1054

Copyright © 2019, Juniper Networks, Inc.

scheduler-maps (For Most Interface Types)

Syntax

```
scheduler-maps {
  map-name {
    forwarding-class class-name scheduler scheduler-name;
  }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify a scheduler map name and associate it with the scheduler configuration and forwarding class.

Options *map-name*—Name of the scheduler map.

The remaining statements are explained separately. See CLI Explorer.

See "Configuring Schedulers" on page 258 .

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

schedulers (CoS)

Syntax

```
schedulers {
  scheduler-name {
    adjust-minimum rate;
    adjust-percent percentage;
    buffer-size (seconds | percent percentage | remainder | temporal microseconds);
    drop-profile-map loss-priority (any | low | medium-low | medium-high | high) protocol
      (any | non-tcp | tcp) drop-profile profile-name;
    excess-priority [ low | medium-low | medium-high | high | none];
    excess-rate (percent percentage | proportion value);
    priority priority-level;
    shaping-rate (percent percentage | rate);
    transmit-rate (percent percentage | rate | remainder) <exact | rate-limit>;
  }
}
```

Hierarchy Level

[edit class-of-service]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series routers.

Description

Specify the scheduler name and parameter values.

Options

scheduler-name—Name of the scheduler to be configured.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- How Schedulers Define Output Queue Properties on page 254
- Default Schedulers Overview on page 257
- Configuring Schedulers on page 258
- Configuring a Scheduler on page 489

Copyright © 2019, Juniper Networks, Inc.

schedulers (Interfaces)

Syntax schedulers *number*;

Hierarchy Level [edit interfaces]

Release Information Statement introduced in Junos OS Release 8.2.

Description Specify number of schedulers for Ethernet IQ2 PIC port interfaces.

Default If you omit this statement, the 1024 schedulers are distributed equally over all ports in multiples of 4.

Options ***number***—Number of schedulers to configure on the port.
Range: 1 through 1024

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring the Number of Schedulers per Port for Ethernet IQ2 PICs on page 803

services (CoS)

Syntax services cos { ... }

Hierarchy Level [edit]

Release Information Statement introduced in Junos OS Release 8.1.

Description Define the service rules to be applied to traffic.

Options **cos**—Identifies the class-of-service set of rules statements.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- *Configuring CoS Rules*

Copyright © 2019, Juniper Networks, Inc.

shaping

Syntax

```
shaping {
  (cbr rate | rtvbr peak rate sustained rate burst length | vbr peak rate sustained rate burst
    length);
  queue-length number;
}
```

Hierarchy Level

```
[edit interfaces interface-name atm-options vpi vpi-identifier],
[edit interfaces interface-name unit logical-unit-number],
[edit interfaces interface-name unit logical-unit-number address address family family
  multipoint-destination address],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number
  address address family family multipoint-destination  address]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For ATM encapsulation only, define the traffic-shaping profile.

For Circuit Emulation PICs, specify traffic shaping in the ingress and egress directions.

For ATM2 IQ interfaces, changing or deleting VP tunnel traffic shaping causes all logical interfaces on a VP to be deleted and then re-added.

VP tunnels are not supported on multipoint interfaces.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Defining Virtual Path Tunnels*
- *Defining the ATM Traffic-Shaping Profile Overview*
- *Configuring ATM QoS or Shaping*
- Applying Scheduler Maps to Logical ATM Interfaces on page 858

Copyright © 2019, Juniper Networks, Inc.

shaping-rate (Applying to an Interface)

Syntax

```
shaping-rate rate;
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name],
[edit class-of-service interfaces interface-name unit logical-unit-number]
```

Release Information

Statement introduced before Junos OS Release 7.4.
[edit class-of-service interfaces *interface-name*] hierarchy level added in Junos OS Release 7.5.
Statement introduced in Junos OS Release 13.2 on PTX Series Packet Transport Routers.
Statement introduced in Junos OS Release 17.3 on PTX10008 Routers.

Description

For logical interfaces on which you configure packet scheduling, configure traffic shaping by specifying the amount of bandwidth to be allocated to the logical interface. Applying a shaping rate can help ensure that higher-priority services do not starve lower-priority services.

For physical interfaces, configure traffic shaping based on the rate-limited bandwidth of the total interface bandwidth.

Logical and physical interface traffic shaping rates are mutually exclusive. This means you can include the **shaping-rate** statement at the **[edit class-of-service interfaces *interface-name*]** hierarchy level or at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level, but not at both.

NOTE: For MX Series routers and for EX Series switches, the shaping rate value for the physical interface at the **[edit class-of-service interfaces *interface-name*]** hierarchy level must be a minimum of 160 Kbps. If the value is less than the sum of the logical interface guaranteed rates, you cannot apply the shaping rate to a physical interface.

For PTX Series routers, the shaping rate value for the physical interface at the **[edit class-of-service interfaces *interface-name*]** hierarchy level must be a minimum of 1 Gbps and an incremental granularity of 0.1 percent of the physical interface speed after that (for example, 10 Mbps increments on a 10 Gbps interface).

For T4000 routers with Type 5 FPCs, the shaping rate value for the physical interface must be a minimum of 292 Kbps. The maximum value of **shaping-rate** is limited by the maximum transmission rate of the interface.

Alternatively, you can configure a shaping rate for a logical interface and oversubscribe the physical interface by including the **shaping-rate** statement at the **[edit class-of-service traffic-control-profiles]** hierarchy level. With this configuration approach, you can

Copyright © 2019, Juniper Networks, Inc.

independently control the delay-buffer rate, as described in "Oversubscribing Interface Bandwidth" on page 273.

For FRF.15 and FRF.16 bundles on link services interfaces, only shaping rates based on percentage are supported.

Default If you do not include this statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level, the default logical interface bandwidth is the average of unused bandwidth for the number of logical interfaces that require default bandwidth treatment. If you do not include this statement at the **[edit class-of-service interfaces *interface-name*]** hierarchy level, the default physical interface bandwidth is the average of unused bandwidth for the number of physical interfaces that require default bandwidth treatment.

Options ***rate***—Peak 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 6,400,000,000,000 bps.

NOTE: Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.

NOTE: For all MX Series and EX Series interfaces, the rate can be from 65,535 to 6,400,000,000,000 bps.

For all PTX Series interfaces, the rate can be from 1,000,000,000 to 160,000,000,000 bps in increments of 0.1 percent of the interface speed.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Applying Scheduler Maps Overview on page 259
- Configuring Virtual LAN Queuing and Shaping on PTX Series Routers on page 631

 Copyright © 2019, Juniper Networks, Inc.

shaping-rate (Schedulers)

Syntax shaping-rate (percent *percentage* | *rate*) <burst-size *bytes*>;

Hierarchy Level [edit class-of-service schedulers *scheduler-name*]

Release Information Statement introduced before Junos OS Release 7.4.
The **burst-size** option added for MIC and MPC interfaces on MX Series routers in Junos OS Release 11.4.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.

Description Define a limit on excess bandwidth usage for a forwarding class/queue.

The **transmit-rate** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level configures the minimum bandwidth allocated to a queue. The transmission bandwidth can be configured as an exact value or allowed to exceed the configured rate if additional bandwidth is available from other queues.

Configure the shaping rate as an absolute maximum usage and not the additional usage beyond the configured transmit rate.

Default If you do not include this statement, the default shaping rate is 100 percent, which is the same as no shaping at all.

Options **percent *percentage***—Shaping rate as a percentage of the available interface bandwidth.
Range: 0 through 100 percent

NOTE: If you configure a shaping rate as a percent in a scheduler, the effective shaping rate is calculated based on the following hierarchy:

1. Logical interface shaping rate, if configured
2. Physical interface shaping rate, if configured
3. Physical interface bandwidth

With SRX300, SRX320, SRX340, SRX345, SRX550m, SRX1500, and vSRX2.0 devices, you can configure both logical interface shaping rates and physical interface shaping rates on the same physical interface. On all other models, you can only configure one or the other on a particular physical interface.

rate—Peak 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).

Copyright © 2019, Juniper Networks, Inc.

Range: 3200 through 6,400,000,000,000 bps

NOTE: Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.

burst-size *bytes*—Maximum burst size, in bytes. The burst value determines the number of rate credits that can accrue when the queue or scheduler node is held in the inactive round robin.

Range: 0 through 1,000,000,000

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Applying Scheduler Maps Overview on page 259

Copyright © 2019, Juniper Networks, Inc.

shaping-rate (Oversubscribing an Interface)

Syntax shaping-rate (percent *percentage* | *rate*) <burst-size *bytes*>;

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 7.6.
Option **burst-size** introduced for Enhanced Queuing (EQ) DPC interfaces on MX Series routers in Junos OS Release 9.4.
Option **burst-size** option introduced for MIC and MPC interfaces on MX Series routers in Junos OS Release 11.4.

NOTE: Option **burst-size** is not supported on MPC5 interfaces. The **burst-size** configuration is allowed on these interfaces, but does not take effect.

Option **burst-size** introduced for IQ2 and IQ2E interfaces in Junos OS Release 12.3.
Statement introduced for PTX Series Packet Transport Routers in Junos OS Release 16.1.
PTX Series Packet Transport Routers do not support the **burst-size** option or defining the **shaping-rate** as a percentage.

Description For Gigabit Ethernet IQ, Channelized IQ PIC, FRF.15 and FRF.16 LSQ interfaces, and for EQ DPC, MIC, and MPC interfaces on MX Series routers, configure a shaping rate for a logical interface. You can also configure an optional burst size for a logical interface on EQ DPC interfaces and on IQ2 and IQ2E PIC interfaces. This can help to ensure that higher-priority services do not starve lower-priority services.

For physical interfaces on T4000 router interfaces on Type 5 FPCs and on PTX Series routers, configure traffic shaping rate.

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).

Default The default behavior depends on various factors. For more information, see Table 28 on page 277.

Options **percent** ***percentage***—For LSQ interfaces, shaping rate as a percentage of the available interface bandwidth.

Range: 1 through 100 percent

rate—For IQ and IQ2 interfaces, and T4000 routers with Type 5 FPCs, peak shaping 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).

Copyright © 2019, Juniper Networks, Inc.

Range:

- IQ and IQ2 interfaces—1000 through 6,400,000,000,000 bps

NOTE: Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.

- T4000 routers with Type 5 FPCs—The shaping rate value for the physical interface must be a minimum of 292 Kbps. The maximum value of **shaping-rate** is limited by the maximum transmission rate of the interface.

burst-size *bytes*—(Optional) Maximum burst size, in bytes.

Range: 0 through 1,000,000,000

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- Oversubscribing Interface Bandwidth on page 273
- output-traffic-control-profile on page 1219

Copyright © 2019, Juniper Networks, Inc.

shaping-rate-excess-high

Syntax shaping-rate-excess-high *rate* [burst-size *bytes*];

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 10.1.

Description For MIC and MPC interfaces on MX Series routers, configure a shaping rate and optional burst size for high-priority excess traffic. This can help to make sure higher priority services do not starve lower priority services.

NOTE: Option burst-size is not supported on MPC5 interfaces. The burst-size configuration is allowed on these interfaces, but does not take effect.

Default If you do not include this statement, the default shaping rate for this priority is determined by the **shaping-rate** statement in the traffic control profile.

Options ***rate***—Peak 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 6,400,000,000,000

NOTE: Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.

Default: None

burst-size *bytes*—Maximum burst size, in bytes.

Range: 0 through 1,000,000,000

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- Oversubscribing Interface Bandwidth on page 273
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

Copyright © 2019, Juniper Networks, Inc.

- shaping-rate-excess-low on page 1273
- shaping-rate-priority-high on page 1277
- shaping-rate-priority-low on page 1279
- shaping-rate-priority-medium on page 1281

Copyright © 2019, Juniper Networks, Inc.

shaping-rate-excess-low

Syntax shaping-rate-excess-low *rate* [burst-size *bytes*];

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 10.1.

Description For MIC and MPC interfaces on MX Series routers, configure a shaping rate and optional burst size for low-priority excess traffic. This can help to make sure higher priority services do not starve lower priority services.

NOTE: Option burst-size is not supported on MPC5 interfaces. The burst-size configuration is allowed on these interfaces, but does not take effect.

Default If you do not include this statement, the default shaping rate for this priority is determined by the **shaping-rate** statement in the traffic control profile.

Options ***rate***—Peak 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 6,400,000,000,000

NOTE: Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.

Default: None

burst-size *bytes*—Maximum burst size, in bytes.

Range: 0 through 1,000,000,000

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- Oversubscribing Interface Bandwidth on page 273
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

Copyright © 2019, Juniper Networks, Inc.

- shaping-rate-excess-high on page 1271
- shaping-rate-priority-high on page 1277
- shaping-rate-priority-low on page 1279
- shaping-rate-priority-medium on page 1281

Copyright © 2019, Juniper Networks, Inc.

shaping-rate-excess-medium-high

Syntax shaping-rate-excess-medium-high *rate* [burst-size *bytes*];

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 16.1.

Description For MPCs (MPC7E-MRATE, MPC7E-10G, MPC8E, and MPC9E) interfaces on MX Series routers, configure a shaping rate and optional burst size for medium-high-priority excess traffic. This can help to make sure higher priority services do not starve lower priority services.

Default If you do not include this statement, the default shaping rate for this priority is determined by the **shaping-rate** statement in the traffic control profile.

Options *rate*—Peak 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 6,400,000,000,000

Default: None

burst-size *bytes*—Maximum burst size, in bytes.

Range: 1 through 1,000,000,000

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- Oversubscribing Interface Bandwidth on page 273
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- shaping-rate-excess-low on page 1273
- shaping-rate-priority-high on page 1277
- shaping-rate-priority-low on page 1279
- shaping-rate-priority-medium on page 1281

Copyright © 2019, Juniper Networks, Inc.

shaping-rate-excess-medium-low

Syntax shaping-rate-excess-medium-low *rate* [burst-size *bytes*];

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 16.1.

Description For MPC7E-MRATE, MPC7E-10G, MPC8E, and MPC9E interfaces on MX Series routers, configure a shaping rate and optional burst size for medium-low-priority excess traffic. This can help to make sure higher priority services do not starve lower priority services.

Default If you do not include this statement, the default shaping rate for this priority is determined by the **shaping-rate** statement in the traffic control profile.

Options ***rate***—Peak 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 6,400,000,000,000

Default: None

burst-size *bytes*—Maximum burst size, in bytes.

Range: 1 through 1,000,000,000

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- enhanced-priority-mode on page 1103
- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- shaping-rate-excess-medium-high on page 1275
- shaping-rate-priority-medium-low on page 1282
- shaping-rate-priority-strict-high on page 1283

Copyright © 2019, Juniper Networks, Inc.

shaping-rate-priority-high

Syntax shaping-rate-priority-high *rate* [burst-size *bytes*];

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 10.1.

Description For MIC and MPC interfaces on MX Series routers, configure a shaping rate and optional burst size for high priority traffic. This can help to make sure higher priority services do not starve lower priority services.

NOTE: Option **burst-size** is not supported on MPC5 interfaces. The **burst-size** configuration is allowed on these interfaces, but does not take effect.

Default If you do not include this statement, the default shaping rate for this priority is determined by the **shaping-rate** statement in the traffic control profile.

Options ***rate***—Peak 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 6,400,000,000,000

NOTE: Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.

Default: None

burst-size *bytes*—Maximum burst size, in bytes.

Range: 0 through 1,000,000,000

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- Oversubscribing Interface Bandwidth on page 273
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

Copyright © 2019, Juniper Networks, Inc.

- shaping-rate-excess-high on page 1271
- shaping-rate-excess-low on page 1273
- shaping-rate-priority-low on page 1279
- shaping-rate-priority-medium on page 1281

Copyright © 2019, Juniper Networks, Inc.

shaping-rate-priority-low

Syntax shaping-rate-priority-low *rate* [burst-size *bytes*];

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 10.1.

Description For MIC and MPC interfaces on MX Series routers, configure a shaping rate and optional burst size for low priority traffic. This can help to make sure higher priority services do not starve lower priority services.

NOTE: Option **burst-size** is not supported on MPC5 interfaces. The **burst-size** configuration is allowed on these interfaces, but does not take effect.

Default If you do not include this statement, the default shaping rate for this priority is determined by the **shaping-rate** statement in the traffic control profile.

Options ***rate***—Peak 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 6,400,000,000,000

NOTE: Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.

Default: None

burst-size *bytes*—Maximum burst size, in bytes.

Range: 0 through 1,000,000,000

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- Oversubscribing Interface Bandwidth on page 273
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

Copyright © 2019, Juniper Networks, Inc.

- shaping-rate-excess-high on page 1271
- shaping-rate-excess-low on page 1273
- shaping-rate-priority-high on page 1277
- shaping-rate-priority-medium on page 1281

Copyright © 2019, Juniper Networks, Inc.

shaping-rate-priority-medium

Syntax shaping-rate-priority-medium *rate* [burst-size *bytes*];

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 10.1.

Description For MIC and MPC interfaces on MX Series routers, configure a shaping rate and optional burst size for medium priority traffic. This can help to make sure higher priority services do not starve lower priority services.

NOTE: Option burst-size is not supported on MPC5 interfaces. The burst-size configuration is allowed on these interfaces, but does not take effect.

Default If you do not include this statement, the default shaping rate for this priority is determined by the **shaping-rate** statement in the traffic control profile.

Options ***rate***—Peak 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 6,400,000,000,000

NOTE: Through Junos OS Release 13.3, the upper limit is 160,000,000,000 bps. Beginning with Junos OS Release 14.1, the upper limit is 6,400,000,000,000 bps.

Default: None

burst-size *bytes*—Maximum burst size, in bytes.

Range: 0 through 1,000,000,000

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- Oversubscribing Interface Bandwidth on page 273
- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260

Copyright © 2019, Juniper Networks, Inc.

- shaping-rate-excess-high on page 1271
- shaping-rate-excess-low on page 1273
- shaping-rate-priority-high on page 1277
- shaping-rate-priority-low on page 1279

shaping-rate-priority-medium-low

Syntax shaping-rate-priority-medium-low *rate* [burst-size *bytes*];

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 16.1.

Description For MPC7E-MRATE, MPC7E-10G, MPC8E, and MPC9E interfaces on MX Series routers, configure a shaping rate and optional burst size for medium-low priority traffic. This can help to make sure higher priority services do not starve lower priority services.

Default If you do not include this statement, the default shaping rate for this priority is determined by the **shaping-rate** statement in the traffic control profile.

Options ***rate***—Peak 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 6,400,000,000,000

Default: None

burst-size *bytes*—Maximum burst size, in bytes.

Range: 1 through 1,000,000,000

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- enhanced-priority-mode on page 1103
- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- shaping-rate-excess-medium-high on page 1275
- shaping-rate-excess-medium-low on page 1276
- shaping-rate-priority-strict-high on page 1283

Copyright © 2019, Juniper Networks, Inc.

shaping-rate-priority-strict-high

Syntax shaping-rate-priority-strict-high *rate* [burst-size *bytes*];

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 16.1.

Description For MPC7E-MRATE, MPC7E-10G, MPC8E, and MPC9E interfaces on MX Series routers, configure a shaping rate and optional burst size for strictly high priority traffic. This can help to make sure higher priority services do not starve lower priority services.

Default If you do not include this statement, the default shaping rate for this priority is determined by the **shaping-rate** statement in the traffic control profile.

Options ***rate***—Peak 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 6,400,000,000,000

Default: None

burst-size *bytes*—Maximum burst size, in bytes.

Range: 1 through 1,000,000,000

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- enhanced-priority-mode on page 1103
- Per-Priority Shaping on MIC and MPC Interfaces Overview on page 955
- shaping-rate-excess-medium-high on page 1275
- shaping-rate-excess-medium-low on page 1276
- shaping-rate-priority-medium-low on page 1282

Copyright © 2019, Juniper Networks, Inc.

shared-bandwidth-policer (Configuring)

Syntax

```
shared-bandwidth-policer;
```

Hierarchy Level

[edit firewall policer *policer-name*],
[edit firewall three-color-policer *policer-name*],
[edit firewall hierarchical-policer *policer-name*]

Release Information

Statement introduced in Junos OS Release 11.2.
Support for MX Series MPC and MIC interfaces added in Junos OS Release 12.1.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Policer instances share bandwidth. This enables configuration of interface-specific policers applied on an aggregated Ethernet bundle or an aggregated SONET bundle to match the effective bandwidth and burst-size to user-configured values. This feature is supported on the following platforms: T Series routers, M120, M10i, M7i (CFEB-E only), M320 (SFPC only), MX240, MX480, and MX960 with DPC, MIC, and MPC interfaces and EX Series switches.

NOTE: This statement is not supported on T4000 Type 5 FPCs.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Policer Support for Aggregated Ethernet Interfaces Overview on page 1017

Copyright © 2019, Juniper Networks, Inc.

shared-instance

Syntax

```
shared-instance instance-name;
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name unit logical-unit-number
  input-traffic-control-profile],
[edit class-of-service interfaces interface-name unit logical-unit-number
  output-traffic-control-profile]
```

Release Information Statement introduced in Junos OS Release 7.6.

Description For Gigabit Ethernet IQ2 and IQ2E PICs only, apply a shared traffic scheduling and shaping profile to the logical interface.

Options ***instance-name***—Name of the shared scheduler and shaper to be applied to this interface

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Shaping on 10-Gigabit Ethernet IQ2 PICs on page 824
- traffic-control-profiles on page 1305

Copyright © 2019, Juniper Networks, Inc.

shared-scheduler

Syntax

```
shared-scheduler;
```

Hierarchy Level

[edit interfaces *interface-name*]

Release Information

Statement introduced in Junos OS Release 7.6.

Description

For Gigabit Ethernet IQ2 PICs only, enable shared schedulers and shapers on this interface. This statement and the **per-unit-scheduler** statement are mutually exclusive. Even so, you can configure one logical interface for each shared instance. This effectively provides the functionality of per-unit scheduling.

For Gigabit Ethernet IQ2 and Ethernet Enhanced IQ2 (IQ2E) PICs on M320 routers, enable shared schedulers on aggregated Ethernet interfaces in link protection mode.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Traffic Control Profiles for Shared Scheduling and Shaping on page 260
- Configuring Shaping on 10-Gigabit Ethernet IQ2 PICs on page 824
- *Configuring Shared Scheduling on Aggregated Ethernet Interfaces*
- traffic-control-profiles on page 1305

Copyright © 2019, Juniper Networks, Inc.

simple-filter (Applying to an Interface)

Syntax

```
simple-filter {
  input filter-name;
}
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number family inet],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number
  family inet]
```

Release Information Statement introduced in Junos OS Release 7.6.

Description Apply a simple filter to an interface. You can apply simple filters to the family **inet** only, and only in the input direction.

Options **input *filter-name***—Name of one filter to evaluate when packets are received on the interface.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Multifield Classifiers on page 98
- filter (Applying to an Interface) on page 1122

Copyright © 2019, Juniper Networks, Inc.

simple-filter

Syntax

```
simple-filter filter-name {
  term term-name {
    from {
      match-conditions;
    }
    then {
      forwarding-class class-name;
      loss-priority (high | low | medium);
    }
  }
}
```

Hierarchy Level

[edit firewall family inet],
[edit logical-systems *logical-system-name* firewall family inet]

Release Information

Statement introduced in Junos OS Release 7.6.
Logical systems support introduced in Junos OS Release 9.3.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Configure simple filters.

Options

filter-name—Name that identifies the simple filter. The name must be a non-reserved string of not more than 64 characters. No special characters are restricted. To include spaces in the name, enclose them in quotation marks (“ ”).

from—Match packet fields to values. If the **from** option is not included, all packets are considered to match and the actions and action modifiers in the **then** statement are taken.

match-conditions—One or more conditions to use to make a match.

term *term-name*—Define a simple-filter term. The name that identifies the term can contain letters, numbers, and hyphens (-), and can be up to 255 characters long. To include spaces in the name, enclose them in quotation marks (“ ”).

then—Actions to take on matching packets. If the **then** option is not included and a packet matches all the conditions in the **from** statement, the packet is accepted.

NOTE: Only **forwarding-class** and **loss-priority** are valid actions in a simple filter configuration.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- simple-filter (Applying to an Interface) on page 1287
- *Simple Filter Overview*
- *How Simple Filters Evaluate Packets*
- *Guidelines for Configuring Simple Filters*
- *Guidelines for Applying Simple Filters*

sip (Application Profile)

Syntax

```
sip {
  video {
    dscp (alias | bits);
    forwarding-class class-name;
  }
  voice {
    dscp (alias | bits);
    forwarding-class class-name;
  }
}
```

Hierarchy Level

[edit services cos application-profile *profile-name*]

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Set the appropriate **dscp** and **forwarding-class** value for SIP traffic.

Default

By default, the system will not alter the DSCP or forwarding class for SIP traffic.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS Rules on Services PICs on page 699
- ftp (Services CoS) on page 1148

Copyright © 2019, Juniper Networks, Inc.

source-address (Services CoS)

Syntax source-address *address*;

Hierarchy Level [edit services cos rule *rule-name* term *term-name* from]

Release Information Statement introduced in Junos OS Release 8.1.
address option enhanced to support IPv4 and IPv6 addresses in Junos OS Release 8.5.

Description Source address for rule matching.

Options ***address***—Source IPv4 or IPv6 address or prefix value.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring CoS Rules on Services PICs on page 699
- *Configuring Match Conditions In CoS Rules*

strict-priority-scheduler

Syntax strict-priority-scheduler;

Hierarchy Level [edit class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 13.3.

Description On PTX Series routers only, enable strict-priority scheduling for all scheduler maps within the traffic control profile.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Example: Configuring Strict-Priority Scheduling on a PTX Series Router on page 637
- Understanding Scheduling on PTX Series Routers on page 604
- Understanding CoS CLI Configuration Statements on PTX Series Routers on page 645
- How Schedulers Define Output Queue Properties on page 254
- **traffic-control-profiles on page 1305**

Copyright © 2019, Juniper Networks, Inc.

sustained-rate

Syntax
sustained-rate *rate*;

Hierarchy Level
[edit class-of-service traffic-control-profiles *traffic-control-profile-name*]
[edit firewall atm-policer*atm-policer-name*]

Release Information
Statement introduced in Junos OS Release 12.1.

Description
(MX Series routers) Define ATM sustained cell rate on ATM MICs in cells per second by entering a decimal number followed by the abbreviation c; where 1 cps = 384 bps.

Options
rate—ATM sustained rate in cells per second.
Range: 61 cps through 353,206 cps.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- *show class-of-service traffic-control-profile*

syslog (Services CoS)

Syntax
syslog;

Hierarchy Level
[edit services cos rule *rule-name* term *term-name* then],
[edit services cos rule *rule-name* term *term-name* then reverse]

Release Information
Statement introduced in Junos OS Release 8.1.

Description
Enable system logging. The system log information from the Multiservices and Services PICs is passed to the kernel for logging in the **/var/log** directory. This setting overrides any **syslog** statement setting included in the service set or interface default configuration.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring CoS Rules on Services PICs on page 699
- *Configuring Actions in CoS Rules*

Copyright © 2019, Juniper Networks, Inc.

system-defaults

Syntax

```
system-defaults {
  classifiers{
    type classifier-name;
  }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information Statement introduced in Junos OS Release 12.2.

Description Define a CoS classifier to support global classifiers.

Options ***classifier-name***—Name of the behavior aggregate (BA) classifier.

type—Traffic type: dscp, dscp-ipv6, or exp.

NOTE: The **dscp** and **dscp-ipv6** classifier types are not supported on ACX Series routers.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS on ACX Series Routers on page 513

Copyright © 2019, Juniper Networks, Inc.

term (Firewall Filter)

Syntax

```
term term-name {
  from {
    match-conditions;
    vxlan {
      vni vni-id
        flags value mask-in-hex value
        reserved1 value
        reserved2 value
      }
    ip-version ipv4 {
      match-conditions-mpls-ipv4-address;
        protocol (tcp | udp) {
          match conditions-mpls-ipv4-port;
        }
    }
  }
  then {
    actions;
  }
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* firewall family *family-name* filter *filter-name*],
[edit firewall family *family-name* filter *filter-name*],
[edit firewall family *family-name* service-filter *filter-name*],
[edit firewall family *family-name* simple-filter *filter-name*],
[edit logical-systems *logical-system-name* firewall family *family-name* filter *filter-name*],
[edit logical-systems *logical-system-name* firewall family *family-name* service-filter *filter-name*],
[edit logical-systems *logical-system-name* firewall family *family-name* simple-filter *filter-name*]

Release Information

Statement introduced before Junos OS Release 7.4.
filter option introduced in Junos OS Release 7.6.
Logical systems support introduced in Junos OS Release 9.3.
ip-version ipv4 support introduced in Junos OS Release 10.1.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Define a firewall filter term.

Options

actions—(Optional) Actions to perform on the packet if conditions match. You can specify one *terminating action* supported for the specified filter type. If you do not specify a terminating action, the packets that match the conditions in the **from** statement are accepted by default. As an option, you can specify one or more *nonterminating actions* supported for the specified filter type.

filter-name—(Optional) For **family *family-name* filter *filter-name*** only, reference another standard stateless firewall filter from within this term.

Copyright © 2019, Juniper Networks, Inc.

from—(Optional) Match packet fields to values. If not included, all packets are considered to match and the actions and action modifiers in the **then** statement are taken.

match-conditions—One or more conditions to use to make a match on a packet.

match-conditions-mpls-ipv4-address—(MPLS-tagged IPv4 traffic only) One or more IP address match conditions to match on the IPv4 packet header. Supports network-based service in a core network with IPv4 packets as an inner payload of an MPLS packet with labels stacked up to five deep.

match-conditions-mpls-ipv4-port—(MPLS-tagged IPv4 traffic only) One or more UDP or TCP port match conditions to use to match a packet in an MPLS flow. Supports network-based service in a core network with IPv4 packets as an inner payload of an MPLS packet with labels stacked up to five deep.

vxlan—(Optional) Match packets belonging to a particular VXLAN Network Identifier (VNI).

term-name—Name that identifies the term. The name can contain letters, numbers, and hyphens (-) and can be up to 64 characters long. To include spaces in the name, enclose it in quotation marks (“ ”).

then—(Optional) Actions to take on matching packets. If not included and a packet matches all the conditions in the **from** statement, the packet is accepted.

The Firewall Filer Match Conditions for the different protocols are explained separately:

- *Firewall Filter Match Conditions for IPv4 Traffic*
- *Firewall Filter Match Conditions for IPv6 Traffic*
- *Firewall Filter Match Conditions for MPLS Traffic*
- *Firewall Filter Match Conditions for MPLS-Tagged IPv4 or IPv6 Traffic*
- *Firewall Filter Match Conditions for VPLS Traffic*
- *Firewall Filter Match Conditions for Protocol-Independent Traffic*
- *Firewall Filter Match Conditions for Protocol-Independent Traffic in Dynamic Service Profiles*
- *Firewall Filter Match Conditions Based on Numbers or Text Aliases*
- *Firewall Filter Match Conditions Based on Bit-Field Values*
- *Firewall Filter Match Conditions Based on Address Fields*
- *Firewall Filter Match Conditions Based on Address Classes*
- *Firewall Filter Match Conditions for Layer 2 Bridging Traffic*
- *Firewall Filter Match Conditions for Layer 2 CCC Traffic*

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

Related Documentation

- *Guidelines for Configuring Firewall Filters*
- Configuring Multifield Classifiers on page 98
- *Guidelines for Configuring Simple Filters*
- *Guidelines for Configuring and Applying Firewall Filters in Logical Systems*

term (Services CoS)

Syntax

```
term term-name {
  from {
    ... from configuration ...
  }
  then {
    ... then configuration ...
  }
}
```

Hierarchy Level

```
[edit services cos rule rule-name]
```

Release Information

Statement introduced in Junos OS Release 8.1.

Description

Define the CoS term properties.

Options

term-name—Identifier for the term.

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS Rules on Services PICs on page 699

Copyright © 2019, Juniper Networks, Inc.

term (Simple Filter)

Syntax

```
term term-name {
  from {
    match-conditions;
  }
  then {
    forwarding-class class-name;
    loss-priority (high | low | medium);
  }
}
```

Hierarchy Level

```
[edit firewall family inet simple-filter filter-name]
```

Release Information Statement introduced in Junos OS Release 7.6.

Description Define a simple filter term.

Options

from—Match packet fields to values. If the **from** option is not included, all packets are considered to match and the actions and action modifiers in the **then** statement are taken.

match-conditions—One or more conditions to use to make a match.

term-name—Name that identifies the term. The name can contain letters, numbers, and hyphens (-), and can be up to 255 characters long. To include spaces in the name, enclose it in quotation marks (“ ”).

then—Actions to take on matching packets. If the **then** option is not included and a packet matches all the conditions in the **from** statement, the packet is accepted. For CoS, only the actions listed are allowed. These statements are explained separately.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- *Multifield Classification*
- *Simple Filter Overview*
- *Firewall Filter Match Conditions for IPv4 Traffic*
- *Firewall Filter Match Conditions for IPv6 Traffic*

Copyright © 2019, Juniper Networks, Inc.

then (Services CoS)

Syntax

```
then {
  application-profileprofile-name;
  dscp (alias | bits);
  forwarding-class class-name;
  syslog;
  reflexive; | revert; | reverse {
    application-profile profile-name;
    dscp (alias | bits);
    forwarding-class class-name;
    syslog;
  }
}
```

Hierarchy Level

```
[edit services cos rule rule-name term term-name]
```

Release Information

Statement introduced in Junos OS Release 8.1.

Description

Define the CoS term actions.

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS Rules on Services PICs on page 699
- *Configuring Actions in CoS Rules*

Copyright © 2019, Juniper Networks, Inc.

three-color-policer (Applying)

Syntax

```
three-color-policer {
  (single-rate | two-rate) policer-name;
}
```

Hierarchy Level

```
[edit firewall family family-name filter filter-name term term-name then]
[edit logical-systems logical-system-name firewall family family-name filter filter-name term
  term-name then]
```

Release Information

Statement introduced in Junos OS Release 7.4.
single-rate statement added in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Apply a tricolor marking policer.

Options

single-rate—Named tricolor policer is a single-rate policer.

two-rate—Named tricolor policer is a two-rate policer.

policer-name—Name of a tricolor policer.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- Configuring and Applying Tricolor Marking Policers on page 175
- *Firewall Filter Nonterminating Actions*
- *Three-Color Policer Configuration Overview*

Copyright © 2019, Juniper Networks, Inc.

three-color-policer (Configuring)

Syntax

```
three-color-policer policer-name | uid {
  action {
    loss-priority high then discard;
  }
  filter-specific;
  logical-interface-policer;
  physical-interface-policer;
  shared-bandwidth-policer;
  single-rate {
    (color-aware | color-blind);
    committed-burst-size bytes;
    committed-information-rate bps;
    excess-burst-size bytes;
  }
  two-rate {
    (color-aware | color-blind);
    committed-burst-size bytes;
    committed-information-rate bps;
    peak-burst-size bytes;
    peak-information-rate bps;
  }
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* firewall],
[edit firewall],
[edit logical-systems *logical-system-name* firewall]

Release Information

Statement introduced before Junos OS Release 7.4.
The **action** and **single-rate** statements added in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
Support at the **[edit dynamic-profiles ... firewall]** hierarchy level introduced in Junos OS Release 11.4.

Description

Configure a three-color policer in static firewall filters or dynamic firewall filters in a dynamic client profile or a dynamic service profile.

Options

policer-name—Name of the three-color policer. Reference this name when you apply the policer to an interface.

uid—When you configure a policer at the **[edit dynamic-profiles]** hierarchy level, you must assign a variable UID as the policer name.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Copyright © 2019, Juniper Networks, Inc.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- Configuring and Applying Tricolor Marking Policers on page 175
- *Three-Color Policer Configuration Guidelines*
- *Basic Single-Rate Three-Color Policers*
- *Basic Two-Rate Three-Color Policers*
- *Two-Color and Three-Color Logical Interface Policers*
- *Two-Color and Three-Color Physical Interface Policers*
- *Two-Color and Three-Color Policers at Layer 2*

Copyright © 2019, Juniper Networks, Inc.

traffic-class (Tunnels)

Syntax

```
traffic-class traffic-class-value;
```

Hierarchy Level

```
[edit interfaces gre-interface name unit unit number tunnel ]
[edit logical-systems logical-system-name interfaces gre-interface-name unit unit number
  tunnel]
```

Release Information Statement introduced before Junos OS Release 11.2R1.

Description Configure a static ToS/Traffic Class value in the IPv4/IPv6 header, respectively, of a GRE tunnel. This can be useful, for example, when you want to apply a different classification to port-mirrored traffic than the source traffic.

This setting overrides ToS reflection set by copying the inner IP header's ToS value to the outer IP packet header.

NOTE: If rewrite rules are configured on the egress WAN interface, those rewrite rules will overwrite this setting. Therefore the **traffic-class** setting only makes sense when no rewrite rules are configured.

Options

traffic-class-value—The value represents the entire 8-bit differentiated services (DS) field in the IP header, and should be chosen based on the desired DSCP/IP precedence value. For example, if a DSCP value of **111000** is desired, then configure the **traffic-class** value to be **224** (corresponding to **111000 00**).

Range: 0-255

Required Privilege Level interface

Related Documentation

- CoS for Tunnels Overview on page 685
- Configuring CoS for GRE and IP-IP Tunnels on page 687
- *Understanding Port Mirroring*

Copyright © 2019, Juniper Networks, Inc.

traffic-class-map

Syntax

```
traffic-class-map {
  dscp traffic-class-map-name{
    traffic-class {
      real-time code-points value;
      network-control code-points value;
      best-effort code-points value;
    }
  }
  exp traffic-class-map-name{
    traffic-class {
      real-time code-points value;
      network-control code-points value;
      best-effort code-points value;
    }
  }
  ieee-802.1 traffic-class-map-name{
    traffic-class {
      real-time code-points value;
      network-control code-points value;
      best-effort code-points value;
    }
  }
  ieee-802.1ad traffic-class-map-name{
    traffic-class {
      real-time code-points value;
      network-control code-points value;
      best-effort code-points value;
    }
  }
  inet precedence traffic-class-map-name{
    traffic-class {
      real-time code-points value;
      network-control code-points value;
      best-effort code-points value;
    }
  }
}
```

Hierarchy Level

[edit class-of-service]

Release Information

Statement introduced in Junos OS Release 14.2 for T4000 routers with Type 5 FPCs.
Statement introduced in Junos OS Release 17.2 for MX Series routers with MPCs.

Description

Define the traffic class map for a packet on the basis of the CoS code points.

Traffic class map is a user-configurable input priority map that helps prioritize and classify traffic entering a Packet Forwarding Engine during ingress oversubscription. You can define traffic class maps for a packet on the basis of the following CoS code points:

Copyright © 2019, Juniper Networks, Inc.

- Differentiated Services code point (DSCP) for IP DiffServ
- IP precedence bits
- MPLS EXP bits
- IEEE 802.1 bits
- IEEE 802.1ad drop eligible indicator (DEI) bits

You can associate the code point values to one of the following traffic classes:

- Real-time
- Network control
- Best-effort

Options

dscp—DSCP traffic class map. Applies to both IPv4 and IPv6 traffic.

exp—MPLS EXP traffic class map.

ieee-802.1—IEEE 802.1 traffic class map.

ieee-802.1ad—IEEE 802.1ad traffic class map.

inet-precedence—IPv4 precedence traffic class map.

traffic-class-map-name—Name of the input priority map.

traffic-class best-effort—Map code points to a best-effort traffic class.

traffic-class network-control—Map code points to a network control traffic class.

traffic-class real-time—Map code points to a real-time traffic class.

code-points *[code-point-alias | code-point-value]*—List of code point aliases and bit strings.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- show class-of-service forwarding-table traffic-class-map on page 1360
- show class-of-service traffic-class-map on page 1409
- Managing Ingress Oversubscription at the PFE on page 660
- Configuring Traffic Class Maps to Manage Ingress Oversubscription on page 661
- Example: Configuring Traffic Class Maps on page 665

Copyright © 2019, Juniper Networks, Inc.

traffic-class-map (Apply to Interface)

Syntax

```
traffic-class-map {
  (dscp | inet precedence) traffic-class-map-name;
  exp traffic-class-map-name;
  (ieee-802.1p | ieee-802.1ad) traffic-class-map-name <vlan-tag (inner | outer)>;
}
```

Hierarchy Level

[edit class-of-service interfaces *interface-name*]

Release Information

Statement introduced in Junos OS Release 14.2 for T4000 routers with Type 5 FPCs.
Statement introduced in Junos OS Release 17.2 for MX Series routers with MPCs.

Description

Associate traffic class maps with an interface.

Options

dscp—DSCP traffic class map; applies to both IPv4 and IPv6 traffic.

exp—MPLS EXP traffic class map.

ieee-802.1—IEEE 802.1 traffic class map.

ieee-802.1ad—IEEE 802.1ad traffic class map.

inet-precedence—IPv4 precedence traffic class map.

traffic-class-map-name—Name of the input priority map.

vlan-tag inner—Traffic class map based on the inner VLAN tag to be associated with an interface. By default, the outer VLAN tag is associated with an interface.

vlan-tag outer—Traffic class map based on the outer VLAN tag to be associated with an interface.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- show class-of-service forwarding-table traffic-class-map on page 1360
- show class-of-service traffic-class-map on page 1409
- Managing Ingress Oversubscription at the PFE on page 660
- Configuring Traffic Class Maps to Manage Ingress Oversubscription on page 661
- Example: Configuring Traffic Class Maps on page 665

Copyright © 2019, Juniper Networks, Inc.

traffic-control-profiles

Syntax

```
traffic-control-profiles profile-name {
  adjust-minimum rate;
  atm-service (cbr | rtvbr | nrtvbr);
  delay-buffer-rate (percent percentage | rate);
  excess-rate (percent percentage  | proportion value );
  excess-rate-high (percent percentage | proportion value);
  excess-rate-low (percent percentage | proportion value);
  guaranteed-rate (percent percentage | rate) <burst-size bytes>;
  max-burst-size cells;
  overhead-accounting (frame-mode | cell-mode | frame-mode-bytes | cell-mode-bytes)
    <bytes (byte-value)>;
  peak-rate rate;
  scheduler-map map-name;
  shaping-rate (percent percentage | rate) <burst-size bytes>;
  shaping-rate-excess-high rate [ burst-size bytes ];
  shaping-rate-excess-low rate [ burst-size bytes ];
  shaping-rate-priority-high rate [ burst-size bytes ];
  shaping-rate-priority-low rate [ burst-size bytes ];
  shaping-rate-priority-medium rate [ burst-size bytes ];
  strict-priority-scheduler;
  sustained-rate rate;
}
```

Hierarchy Level

[edit class-of-service]

Release Information

Statement introduced in Junos OS Release 7.6.

Description

For Gigabit Ethernet IQ, Channelized IQ PICs, FRF.15 and FRF.16 LSQ interfaces, Enhanced Queuing (EQ) DPCs, and PTX Series routers only, configure traffic shaping and scheduling profiles. For Enhanced EQ PICs, EQ DPCs, and PTX Series routers only, you can include the **excess-rate** statement.

Options

profile-name—Name of the traffic-control profile.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Oversubscribing Interface Bandwidth on page 273
- Understanding Scheduling on PTX Series Routers on page 604

Copyright © 2019, Juniper Networks, Inc.

traffic-manager

List of Syntax

Syntax (MX Series, PTX Series) Configure Queue Monitoring on page 1306
Syntax (MX Series, T Series) on page 1306
Syntax (M Series) on page 1306
Syntax (QFX Series) on page 1307
Syntax (vSRX) on page 1307

Syntax (MX Series, PTX Series) Configure Queue Monitoring

```
traffic-manager {
  egress-shaping-overhead number;
  ingress-shaping-overhead number;
  mode {
    egress-only;
    ingress-and-egress;
    session-shaping;
  }
  enhanced-priority-mode;
  no-enhanced-priority-mode;
  packet-timestamp {
    enable;
  }
  queue-threshold {
    fabric-queue {
      priority high/low {
        threshold threshold-percentage;
      }
    }
    wan-queue {
      priority high/medium-high/medium-low/low {
        threshold threshold-percentage;
      }
    }
  }
}
```

Syntax (MX Series, T Series)

```
traffic-manager {
  egress-shaping-overhead number;
  ingress-shaping-overhead number;
  mode {
    egress-only;
    ingress-and-egress;
  }
}
```

Syntax (M Series)

```
traffic-manager {
  egress-shaping-overhead number;
  ingress-shaping-overhead number;
  mode {
    egress-only;
    ingress-and-egress;
```


Copyright © 2019, Juniper Networks, Inc.

```
    session-shaping;
  }
}
```

Syntax (QFX Series)

```
traffic-manager {
  buffer-monitor-enable;
  packet-timestamp {
    enable;
  }
  queue-threshold {
    fabric-queue {
      priority high/low {
        threshold threshold-percentage;
      }
    }
    wan-queue {
      priority high/medium-high/medium-low/low {
        threshold threshold-percentage;
      }
    }
  }
}
```

Syntax (vSRX)

```
traffic-manager {
  egress-shaping-overhead number;
}
```

Hierarchy Level

[edit chassis fpc *slot-number*],
[edit chassis fpc *slot-number* pic *pic-number*],
[edit chassis lcc *number* fpc *slot-number* pic *pic-number*] (Routing Matrix)

Release Information

Statement introduced in Junos OS Release 8.3.

Description

Enable CoS queuing, scheduling, and shaping on an L2TP session.

NOTE: Committing changes to **traffic-manager** automatically restarts any necessary components (PICs, DPCs, or FPCs).

Options

buffer-monitor-enable—QFX5000 Series only. Enable port buffer monitoring. Buffer utilization data is collected in one-second intervals and compared with the data from the previous interval. The larger value is kept to keep track of peak buffer occupancy for each queue or priority group.

queue-threshold—Enable monitoring of Fabric and WAN queues. When the **fabric-queue** statement is configured, an SNMP trap is generated whenever the fabric power utilization exceeds the configured threshold value.

When **wan-queue** is configured, an SNMP trap is generated whenever the WAN queue depth exceeds the configured threshold value.

egress-shaping-overhead ***number***—When traffic management (queueing and scheduling) is configured on the egress side, the number of CoS shaping overhead bytes to add to the packets on the egress interface.

Replace ***number*** with a value from **-63** through **192** bytes.

For vSRX, replace ***number*** with a value from **-62** through **192** bytes.

NOTE: The L2 headers (DA/SA + VLAN tags) are automatically a part of the shaping calculation.

ingress-shaping-overhead ***number***—When L2TP session shaping is configured, the number of CoS shaping overhead bytes to add to the packets on the ingress side of the L2TP tunnel to determine the shaped session packet length.

When session shaping is not configured and traffic management (queueing and scheduling) is configured on the ingress side, the number of CoS shaping overhead bytes to add to the packets on the ingress interface.

Replace ***number*** with a value from **—63** through **192** bytes.

mode—Configure CoS traffic manager mode of operation. This option has the following suboptions:

- **egress-only**—Enable CoS queuing and scheduling on the egress side for the PIC that houses the interface. This is the default mode for an Enhanced Queueing (EQ) DPC on MX Series routers.

NOTE: If ingress packet drops are observed at a high rate for an IQ2 or IQ2E PIC, configure the **traffic-manager** statement to work in the **egress-only** mode.

- **ingress-and-egress**—Enable CoS queueing and scheduling on both the egress and ingress sides for the PIC. This is the default mode for IQ2 and IQ2E PICs on M Series and T Series routers.

Copyright © 2019, Juniper Networks, Inc.

NOTE:

- For EQ DPCs, you must configure the **traffic-manager** statement with **ingress-and-egress** mode to enable ingress CoS on the EQ DPC.
- EQ DPCs have 250 ms of buffering, with only egress queueing (default mode). When **ingress-and-egress** is configured, the buffer is partitioned as 50 ms for the ingress direction and 200 ms for the egress direction.

- **session-shaping**—(M Series routers only) Configure the IQ2 PIC mode for session-aware traffic shaping to enable L2TP session shaping.

enhanced-priority-mode—Enable the enhanced priority mode. When you enable the enhanced priority mode, the scheduler supports four additional per-priority shaping rates and two additional excess priorities at the interface and interface set level. The four additional per-priority shaping rates are: Guaranteed Strict-high, Guaranteed Medium-low, Excess medium-high, and Excess medium-low. The two additional excess priorities are: Excess-rate Medium- high and Excess-rate Medium-low. This is the default mode for PTX Series routers.

no-enhanced-priority-mode—Disable the enhanced priority mode. This is the default mode for MX Series routers.

NOTE: **The line card reboots when you enable or disable the enhanced priority mode feature.**

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS for L2TP Tunnels on ATM Interfaces on page 848
- Enabling a Timestamp for Ingress and Egress Queue Packets on page 594
- **show interfaces queue on page 1421**

Copyright © 2019, Juniper Networks, Inc.

translation-table

Syntax

```
translation-table {
  (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 | to-exp-from-exp |
    to-inet-precedence-from-inet-precedence) table-name {
    to-code-point value from-code-points (* | [ values ]);
  }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 9.3.
Support on Multiservices PIC added in Junos OS Release 9.5.

Description

For an Enhanced IQ PIC or Multiservices PIC, specify the input translation tables. You must also apply the translation table to a logical interface on the Enhanced IQ PIC or Multiservices PIC.

Default

If you do not include this statement, the ToS bit values in received packet headers are not changed by the PIC.

Options

to-dscp-from-dscp—(Optional) Translate incoming IPv4 DSCP values to new values. You must also configure and apply a DSCP classifier.

to-dscp-ipv6-from-dscp-ipv6—(Optional) Translate incoming IPv6 DSCP values to new values. You must also configure and apply an IPv6 DSCP classifier.

to-inet-precedence-from-inet-precedence—(Optional) Translate incoming INET precedence values to new values.

to-exp-from-exp—(Optional) Translate incoming MPLS EXP values to new values.

table-name—The name of the translation table.

value—The bit string to which to translate the incoming bit value.

value(s)—The bit string(s) from which the incoming bit value(s) are translated.

*****—(Optional) This translation matches all bit patterns not explicitly listed.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring ToS Translation Tables on page 745
- Multiservices PIC ToS Translation on page 707

Copyright © 2019, Juniper Networks, Inc.

transmit-rate (Schedulers)

Syntax transmit-rate (*rate* | percent *percentage* | remainder) <exact | rate-limit>;

Hierarchy Level [edit class-of-service schedulers *scheduler-name*]

Release Information Statement introduced before Junos OS Release 7.4.
rate-limit option introduced in Junos OS Release 8.3. Applied to the Multiservices PICs in Junos OS Release 9.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series routers.
Statement introduced in Junos OS Release 12.2 for ACX Series routers.

Description Specify the transmit rate or percentage for a scheduler.

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, respectively.

Options

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. This value should never exceed the rate-controlled amount. For PTX Series routers, this option is allowed only on the non-strict-high (high, medium-high, medium-low, or low) queues.

percent *percentage*—Percentage of transmission capacity. A percentage of zero drops all packets in the queue unless additional bandwidth is available from other queues.

Range: 0 through 100 percent for M, MX and T Series routers and EX Series switches; 1 through 100 percent for PTX Series routers; 0 through 200 percent for the SONET/SDH OC48/STM16 IQE PIC

NOTE:

- On M Series Multiservice Edge Routers, for interfaces configured on 4-port E1 and 4-port T1 PICs only, you can configure a ***percentage*** value only from 11 through 100. These two PICs do not support transmission rates less than 11 percent.
- The configuration of the **transmit-rate percent 0 exact** statement at the [**edit class-of-service schedules** *scheduler-name*] hierarchy is ineffective on T4000 routers with Type 5 FPC.
- On MIC and MPC interfaces on MX Series routers, when the transmit rate is configured as a percentage and **exact** or **rate-limit** is enabled on a queue, the shaping rate of the parent node is used to compute the transmit rate. If **exact** or **rate-limit** is not configured, the guaranteed rate of the parent node is used to compute the transmit rate.

Copyright © 2019, Juniper Networks, Inc.

- On PTX Series routers, unconfigured interfaces are equivalent to **percent 0**. This means the system offers no guaranteed rate on the interface, and the queue will always be scheduled in the excess priority.

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 6,400,000,000,000 bps

NOTE: For all MX Series interfaces, the rate can be from 65,535 through 6,400,000,000,000 bps.

rate-limit—(Optional) Limit the transmission rate to the rate-controlled amount by applying a policing action to the queue. Packets are hard-dropped when traffic exceeds the specified maximum transmission rate.

NOTE: For PTX Series routers, this option is allowed only on the strict-high queue. We recommend that you configure rate limit on strict-high queues because the other queues may not meet their guaranteed bandwidths. The **rate-limit** option cannot rate limit the queue if strict-priority scheduling is configured with the **strict-priority-scheduler** statement.

NOTE: The configuration of the **rate-limit** statement is supported on T4000 routers only with a Type 5 FPC.

remainder—Use the remaining rate available.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Schedulers on page 258
- Configuring Scheduler Transmission Rate on page 282
- Understanding Scheduling on PTX Series Routers on page 604

Copyright © 2019, Juniper Networks, Inc.

transmit-weight

Syntax
```
transmit-weight (cells number | percent number);
```

Hierarchy Level
```
[edit interfaces at-fpc/pic/portatm-options scheduler-maps map-name forwarding-class)
  class-name]
```

Release Information Statement introduced before Junos OS Release 7.4.

Description For ATM2 IQ interfaces only, assign a transmission weight to a forwarding class.

Default 95 percent for queue 0, 5 percent for queue 3.

Options

percent *percentage*—Transmission weight of the forwarding class as a percentage of the total bandwidth.

Range: 5 through 100

cells *number*—Transmission weight of the forwarding class as a number of cells.

Range: 0 through 32,000

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Applying Scheduler Maps to ATM Interfaces on page 853

transparent

Syntax
```
transparent;
```

Hierarchy Level
```
[edit class-of-service interfaces interface-name unit logical-unit-number classifiers ieee802.1
  vlan-tag]
```

Release Information Statement introduced in Junos OS Release 11.2

Description Packet classification based on the transparent VLAN tag.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Copyright © 2019, Juniper Networks, Inc.

tri-color

Syntax

```
tri-color;
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information Statement introduced in Junos OS Release 7.4.

Description For IPv4 packets on M320, MX Series, T Series routers with Enhanced II Flexible PIC Concentrators (FPCs), and EX Series switches, enable tricolor marking (TCM), as defined in RFC 2698.

Default If you do not include this statement, tricolor marking is not enabled and the medium packet loss priority (PLP) is not configurable.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring and Applying Tricolor Marking Policers on page 175

Copyright © 2019, Juniper Networks, Inc.

tunnel

Syntax

```
tunnel {
  allow-fragmentation;
  backup-destination address;
  destination destination-address;
  do-not-fragment;
  key number;
  routing-instance {
    destination routing-instance-name;
  }
  source source-address;
  traffic-class traffic-class-value;
  ttl number;
}
```

Hierarchy Level

[edit interfaces *interface-name* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1 for EX Series switches.

Description

Configure a tunnel. You can use the tunnel for unicast and multicast traffic or just for multicast traffic. You can also use tunnels for encrypted traffic or virtual private networks (VPNs).

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Configuring Encryption Interfaces*

Copyright © 2019, Juniper Networks, Inc.

tunnel-services (Chassis)

Syntax

```
tunnel-services {
  bandwidth bandwidth-value;
  tunnel-only;
}
```

Hierarchy Level

```
[edit chassis fpc slot-number pic number]
```

Release Information

Statement introduced in Junos OS Release 8.2.
Statement introduced in Junos OS Release 12.3X54 for ACX Series routers.

Description

For MX Series 5G Universal Routing Platforms, configure the amount of bandwidth for tunnel services.

For ACX Series routers, configure the amount of bandwidth for tunnel services. Only bandwidths of 1 Gbps and 10 Gbps are supported for ACX routers.

For M7i, M10i, M120, M320, T Series and TX Matrix routers with IQ2 PICs and IQ2E PICs, configure support for per unit scheduling for GRE tunnels. You can specify the IQ2 and IQ2E PICs to work exclusively in tunnel mode or as a regular PIC. The default setting uses IQ2 and IQ2E PICs as a regular PIC. If you do not configure the **tunnel-only** option, the IQ2 and IQ2 PICs operate as regular PICs. For M7i, M10i, M120, M320, T Series and TX Matrix routers with IQ2 PICs and IQ2E PICs, you can use the **tunnel-only** option to specify that an IQ2 or IQ2E PIC work in tunnel mode only.

NOTE: Bandwidth rates of 20 gigabits per second and 40 gigabits per second require use of an MX Series router with the 100-Gigabit Ethernet Modular Port Concentrator (MPC) and the 100-Gigabit CFP MIC.

NOTE: On MX80 routers and MX Series routers with Trio-based FPCs, when ingress queuing is enabled for a PIC, tunnel services and inline services are not supported on the same PIC.

Options

tunnel–only (Optional)—For M7i, M10i, M120, M320, T Series and TX Matrix routers with IQ2 PICs and IQ2E PICs, specify that an IQ2 or IQ2E PIC work in tunnel mode only.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Copyright © 2019, Juniper Networks, Inc.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Example: Configuring Tunnel Interfaces on a Gigabit Ethernet 40-Port DPC*
- *Example: Configuring Tunnel Interfaces on a 10-Gigabit Ethernet 4-Port DPC*
- *Example: Configuring Tunnel Interfaces on the MPC3E*
- *bandwidth (Tunnel Services)*

Copyright © 2019, Juniper Networks, Inc.

unit

Syntax

```
unit logical-unit-number {
  classifiers {
    type (classifier-name | default) family (mpls | all);
  }
  forwarding-class class-name;
  fragmentation-map map-name;
  input-traffic-control-profile profile-name shared-instance instance-name;
  output-traffic-control-profile profile-name shared-instance instance-name;
  per-session-scheduler;
  rewrite-rules {
    dscp (rewrite-name | default);
    dscp-ipv6 (rewrite-name | default);
    exp (rewrite-name | default)protocol protocol-types;
    exp-push-push-push default;
    exp-swap-push-push default;
    ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
    inet-precedence (rewrite-name | default);
  }
  scheduler-map map-name;
  shaping-rate rate;
}
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.

Options

logical-unit-number—Number of the logical unit.

Range: 0 through 16,384

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 36
- Configuring Rewrite Rules on page 391

Copyright © 2019, Juniper Networks, Inc.

vbr

Syntax

vbr peak *rate* sustained *rate* burst *length*;

Hierarchy Level

[edit interfaces *interface-name* atm-options vpi *vpi-identifier* shaping],
[edit interfaces *interface-name* unit *logical-unit-number* address *address* family *family* multipoint-destination *address* shaping],
[edit interfaces *interface-name* unit *logical-unit-number* shaping],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* address *address* family *family* multipoint-destination *address* shaping],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* shaping]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For ATM encapsulation only, define the variable bandwidth utilization in the traffic-shaping profile.

When you configure the variable bandwidth utilization, you must specify all three options (**burst**, **peak**, and **sustained**). You can specify the rate 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). You can also specify the rate in cells per second by entering a decimal number followed by the abbreviation **c**; values expressed in cells per second are converted to bits per second by means of the formula 1 cps = 384 bps.

Default

If the **vbr** statement is not specified, bandwidth utilization is unlimited.

Options

burst *length*—Burst length, in cells. If you set the length to 1, the peak traffic rate is used.

Range: 1 through 4000 cells

peak *rate*—Peak rate, in bits per second or cells per second.

Range: For ATM1 interfaces, 33 Kbps through 135.6 Mbps (ATM OC3); 33 Kbps through 276 Mbps (ATM OC12). For ATM2 IQ OC3 and OC12 interfaces, 33 Kbps through 542,526,792 bps. For ATM2 IQ OC48 interfaces, 33 Kbps through 2,170,107,168 bps. For ATM2 IQ DS3 and E3 interfaces, from 33 Kbps through the maximum rate, which depends on the ATM encapsulation and framing you configure.

sustained *rate*—Sustained rate, in bits per second or cells per second.

Range: For ATM1 interfaces, 33 Kbps through 135.6 Mbps (ATM OC3); 33 Kbps through 276 Mbps (ATM OC12). For ATM2 IQ OC3 and OC12 interfaces, 33 Kbps through 542,526,792 bps. For ATM2 IQ OC48 interfaces, 33 Kbps through 2,170,107,168 bps. For ATM2 IQ DS3 and E3 interfaces, from 33 Kbps through the maximum rate, which depends on the ATM encapsulation and framing you configure.

Copyright © 2019, Juniper Networks, Inc.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- *Configuring ATM CBR*
- Applying Scheduler Maps to Logical ATM Interfaces on page 858
- cbr on page 1060
- rtvbr on page 1253
- shaping on page 1264

vc-cos-mode

Syntax
```
vc-cos-mode (alternate | strict);
```

Hierarchy Level
[edit interfaces *interface-name* atm-options scheduler-maps *map-name*]

Release Information
Statement introduced before Junos OS Release 7.4.

Description
For ATM2 IQ interfaces only, specify packet-scheduling priority value for ATM2 IQ VC tunnels.

Options
alternate—VC CoS queue has high priority. The scheduling of the queues alternates between the high-priority queue and the remaining queues, so every other scheduled packet is from the high-priority queue.

strict—VC CoS queue has strictly high priority. A queue with strict high priority is always scheduled before the remaining queues. The remaining queues are scheduled in round-robin fashion.

Default: alternate

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- *ATM2 IQ VC Tunnel CoS Components Overview*
- Applying Scheduler Maps to ATM Interfaces on page 853

Copyright © 2019, Juniper Networks, Inc.

vci

Syntax

vci *vpi-identifier.vci-identifier*;

Hierarchy Level

[edit interfaces at-*fpc/pic/port* unit *logical-unit-number*],
[edit interfaces at-*fpc/pic/port* unit *logical-unit-number* family *family* address *address* multipoint-destination *address*],
[edit logical-systems *logical-system-name* interfaces at-*fpc/pic/port* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces at-*fpc/pic/port* unit *logical-unit-number* family *family* address *address* multipoint-destination *address*]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.

Description

For ATM point-to-point logical interfaces only, configure the virtual circuit identifier (VCI) and virtual path identifier (VPI).

To configure a VPI for a point-to-multipoint interface, specify the VPI in the *multipoint-destination* statement.

VCIs 0 through 31 are reserved for specific ATM values designated by the ATM Forum.

Options

vci-identifier—ATM virtual circuit identifier. Unless you configure the interface to use promiscuous mode, this value cannot exceed the highest-numbered VC configured for the interface with the **maximum-vcs** option of the **vpi** statement.

Range: 0 through 4089 or 0 through 65,535 with promiscuous mode, with VCIs 0 through 31 reserved.

vpi-identifier—ATM virtual path identifier.

Range: 0 through 255

Default: 0

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Configuring a Point-to-Point ATM1 or ATM2 IQ Connection*
- Applying Scheduler Maps to Logical ATM Interfaces on page 858

Copyright © 2019, Juniper Networks, Inc.

video (Application Profile)

Syntax

```
video {
  dscp (alias | bits);
  forwarding-class class-name;
}
```

Hierarchy Level

[edit services cos application-profile*profile-name* sip]

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Set the appropriate **dscp** and **forwarding-class** values for SIP video traffic.

Default

By default, the system will not alter the DSCP or forwarding class for SIP video traffic.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

-
- voice (Application Profile) on page 1326

Copyright © 2019, Juniper Networks, Inc.

vlan-tag

Syntax

```
vlan-tag (outer | outer-and-inner);
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules
  ieee-802.1 (rewrite-name | default)]
```

Release Information Statement introduced in Junos OS Release 8.1.

Description Apply this IEEE-802.1 rewrite rule to the outer VLAN tag or, if available, both outer and inner VLAN tags.

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

- Applying IEEE 802.1p Rewrite Rules to Dual VLAN Tags on page 393

Copyright © 2019, Juniper Networks, Inc.

vlan-tagging

Syntax
```
vlan-tagging;
```

Syntax (QFX Series, NFX Series, and EX4600)
```
vlan-tagging;
```

Syntax (SRX Series Interfaces)
```
vlan-tagging native-vlan-id vlan-id;
```

Hierarchy Level
```
[edit interfaces interface-name],
[edit logical-systems logical-system-name interfaces interface-name]
```

QFX Series, NFX Series, and EX4600 Interfaces
```
[edit interfaces (QFX Series) interface-name ]
[edit interfaces (QFX Series) interface-range interface-range-name ]
```

SRX Series Interfaces
```
[edit interfaces interface ]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 9.5.
Statement introduced in Junos OS Release 11.3 for the QFX Series.
Statement introduced in Junos OS Release 12.2 for ACX Series Universal Metro Routers.
Statement introduced in Junos OS Release 13.2 for PTX Series Routers.
Statement introduced in Junos OS Release 14.1X53-D10 for the QFX Series.

Description

For Fast Ethernet and Gigabit Ethernet interfaces, aggregated Ethernet interfaces configured for VPLS, and pseudowire subscriber interfaces, enable the reception and transmission of 802.1Q VLAN-tagged frames on the interface.

NOTE: For QFX Series configure VLAN identifier for untagged packets received on the physical interface of a trunk mode interface. Enable VLAN tagging. The platform receives and forwards single-tag frames with 802.1Q VLAN tags.

On EX Series switches except for EX4300 and EX9200 switches, the **vlan-tagging** and **family ethernet-switching** statements cannot be configured on the same interface. Interfaces on EX2200, EX3200, EX3300, EX4200, and EX4500 switches are set to **family ethernet-switching** by the default factory configuration. EX6200 and EX8200 switch interfaces do not have a default **family** setting.

Copyright © 2019, Juniper Networks, Inc.

Default VLAN tagging is disabled by default.

Options **native-vlan-id**— (SRX Series)Configures a VLAN identifier for untagged packets. Enter a number from 0 through 4094.

NOTE: The **native-vlan-id** can be configured only when either **flexible-vlan-tagging** mode or **interface-mode** trunk is configured.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *802.1Q VLANs Overview*
- *Configuring a Layer 3 Subinterface (CLI Procedure)*
- *Configuring Tagged Aggregated Ethernet Interfaces*
- *Example: Configuring Layer 3 Subinterfaces for a Distribution Switch and an Access Switch*
- *vlan-id*
- *Configuring a Layer 3 Logical Interface*
- *Configuring VLAN Tagging*

Copyright © 2019, Juniper Networks, Inc.

voice (Application Profile)

Syntax

```
voice {
  dscp (alias | bits);
  forwarding-class class-name;
}
```

Hierarchy Level

[edit services cos application-profile*profile-name* sip]

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Set the appropriate **dscp** and **forwarding-class** values for SIP voice traffic.

Default

By default, the system will not alter the DSCP or forwarding class for SIP voice traffic.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS Rules on Services PICs on page 699
- video (Application Profile) on page 1322

Copyright © 2019, Juniper Networks, Inc.

CHAPTER 29

Operational Commands

- show class-of-service classifier
- show class-of-service code-point-aliases
- show class-of-service drop-profile
- show class-of-service fabric scheduler-map
- show class-of-service fabric statistics
- show class-of-service forwarding-table
- show class-of-service forwarding-table classifier
- show class-of-service forwarding-table classifier mapping
- show class-of-service forwarding-table drop-profile
- show class-of-service forwarding-table fabric scheduler-map
- show class-of-service forwarding-table rewrite-rule
- show class-of-service forwarding-table rewrite-rule mapping
- show class-of-service forwarding-table scheduler-map
- show class-of-service forwarding-table traffic-class-map
- show class-of-service fragmentation-map
- show class-of-service interface
- show class-of-service loss-priority-rewrite
- show class-of-service l2tp-session
- show class-of-service policy-map
- show class-of-service rewrite-rule
- show class-of-service routing-instance
- show class-of-service scheduler-hierarchy interface
- show class-of-service scheduler-map
- show class-of-service traffic-class-map
- show class-of-service translation-table
- show interfaces forwarding-class-counters
- show interfaces queue
- show interfaces voq

Copyright © 2019, Juniper Networks, Inc.

show class-of-service classifier

Syntax

```
show class-of-service classifier
<name name>
<type dscp | type dscp-ipv6 | type exp | type ieee-802.1 | type inet-precedence>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

For each class-of-service (CoS) classifier, display the mapping of code point value to forwarding class and loss priority.

Options

none—Display all classifiers.

name *name*—(Optional) Display named classifier.

type dscp—(Optional) Display all classifiers of the Differentiated Services code point (DSCP) type.

type dscp-ipv6—(Optional) Display all classifiers of the DSCP for IPv6 type.

type exp—(Optional) Display all classifiers of the MPLS experimental (EXP) type.

type ieee-802.1—(Optional) Display all classifiers of the ieee-802.1 type.

type inet-precedence—(Optional) Display all classifiers of the inet-precedence type.

Required Privilege Level

view

List of Sample Output

show class-of-service classifier type ieee-802.1 on page 1329
show class-of-service classifier type ieee-802.1 (QFX Series) on page 1329

Output Fields

Table 148 on page 1328 describes the output fields for the **show class-of-service classifier** command. Output fields are listed in the approximate order in which they appear.

Table 148: show class-of-service classifier Output Fields

Field Name	Field Description
Classifier	Name of the classifier.
Code point type	Type of the classifier: **exp** (not on EX Series switch), **dscp**, **dscp-ipv6** (not on EX Series switch), **ieee-802.1**, or **inet-precedence**.
Index	Internal index of the classifier.

Copyright © 2019, Juniper Networks, Inc.

Table 148: show class-of-service classifier Output Fields (continued)

Field Name	Field Description
Code point	Code point value used for classification
Forwarding class	Classification of a packet affecting the forwarding, scheduling, and marking policies applied as the packet transits the router.
Loss priority	Loss priority value used for classification. For most platforms, the value is **high** or **low**. For some platforms, the value is **high**, **medium-high**, **medium-low**, or **low**.

Sample Output

show class-of-service classifier type ieee-802.1

user@host> show class-of-service classifier type ieee-802.1

```
Classifier: ieee802.1-default, Code point type: ieee-802.1, Index: 3
Code Point         Forwarding Class                  Loss priority
  000              best-effort                         low
  001              best-effort                         high
  010              expedited-forwarding                low
  011              expedited-forwarding                high
  100              assured-forwarding                  low
  101              assured-forwarding                  medium-high
  110              network-control                     low
  111              network-control                     high

Classifier: users-ieee802.1, Code point type: ieee-802.1
  Code point         Forwarding class                  Loss priority
  100                expedited-forwarding              low
```

show class-of-service classifier type ieee-802.1 (QFX Series)

user@switch> show class-of-service classifier type ieee-802.1

```
Classifier: ieee8021p-default, Code point type: ieee-802.1, Index: 11
  Code point         Forwarding class                  Loss priority
  000                best-effort                       low
  001                best-effort                       low
  010                best-effort                       low
  011                fcoe                              low
  100                no-loss                           low
  101                best-effort                       low
  110                network-control                   low
  111                network-control                   low

Classifier: ieee8021p-untrust, Code point type: ieee-802.1, Index: 16
  Code point         Forwarding class                  Loss priority
  000                best-effort                       low
  001                best-effort                       low
  010                best-effort                       low
  011                best-effort                       low
  100                best-effort                       low
  101                best-effort                       low
  110                best-effort                       low
```

Copyright © 2019, Juniper Networks, Inc.

```
  111               best-effort                    low

Classifier: ieee-mcast, Code point type: ieee-802.1, Index: 46
  Code point        Forwarding class               Loss priority
  000               mcast                          low
  001               mcast                          low
  010               mcast                          low
  011               mcast                          low
  100               mcast                          low
  101               mcast                          low
  110               mcast                          low
  111               mcast                          low
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service code-point-aliases

Syntax

```
show class-of-service code-point-aliases
<dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display the mapping of class-of-service (CoS) code point aliases to corresponding bit patterns.

Options

none—Display code point aliases of all code point types.

dscp—(Optional) Display Differentiated Services code point (DSCP) aliases.

dscp-ipv6—(Optional) Display IPv6 DSCP aliases.

exp—(Optional) Display MPLS EXP code point aliases.

ieee-802.1—(Optional) Display IEEE-802.1 code point aliases.

inet-precedence—(Optional) Display IPv4 precedence code point aliases.

Required Privilege Level

view

List of Sample Output

show class-of-service code-point-aliases exp on page 1332

Output Fields

Table 149 on page 1331 describes the output fields for the **show class-of-service code-point-aliases** command. Output fields are listed in the approximate order in which they appear.

Table 149: show class-of-service code-point-aliases Output Fields

Field Name	Field Description
Code point type	Type of the code points displayed: **dscp**, **dscp-ipv6** (not on EX Series switch), **exp** (not on EX Series switch or the QFX Series), **ieee-802.1**, or **inet-precedence** (not on the QFX Series).
Alias	Alias for a bit pattern.
Bit pattern	Bit pattern for which the alias is displayed.

Copyright © 2019, Juniper Networks, Inc.

Sample Output

show class-of-service code-point-aliases exp

```
user@host> show class-of-service code-point-aliases exp
```

```
Code point type: exp
  Alias              Bit pattern
  af11               100
  af12               101
  be                 000
  be1                001
  cs6                110
  cs7                111
  ef                 010
  ef1                011
  nc1                110
  nc2                111
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service drop-profile

Syntax

```
show class-of-service drop-profile
<profile-name profile-name>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display data points for each class-of-service (CoS) random early detection (RED) drop profile.

Options

none—Display all drop profiles.

profile-name *profile-name*—(Optional) Display the specified profile only.

Required Privilege Level

view

List of Sample Output

show class-of-service drop-profile on page 1334
show class-of-service drop-profile (EX4200 Switch) on page 1334
show class-of-service drop-profile (EX8200 Switch) on page 1334

Output Fields

Table 150 on page 1333 describes the output fields for the **show class-of-service drop-profile** command. Output fields are listed in the approximate order in which they appear.

Table 150: show class-of-service drop-profile Output Fields

Field Name	Field Description
Drop profile	Name of a drop profile.
Type	Type of drop profile: • **discrete** (default) • **interpolated** (EX8200 switches, QFX Series switches, QFabric systems, EX4600 switches, OCX Series switches only)
Index	Internal index of this drop profile.
Fill Level	Percentage fullness of a queue.
Drop probability	Drop probability at this fill level.

Copyright © 2019, Juniper Networks, Inc.

Sample Output

show class-of-service drop-profile

```
user@host> show class-of-service drop-profile
Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: user-drop-profile, Type: interpolated, Index: 2989
  Fill level    Drop probability
           0                   0
           1                   1
           2                   2
           4                   4
           5                   5
           6                   6
           8                   8
          10                  10
          12                  15
          14                  20
          15                  23
        ... 64 entries total
          90                  96
          92                  96
          94                  97
          95                  98
          96                  98
          98                  99
          99                  99
         100                 100
```

show class-of-service drop-profile (EX4200 Switch)

```
user@switch> show class-of-service drop-profile
Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level
         100
Drop profile: dp1, Type: discrete, Index: 40496
  Fill level
          10
```

show class-of-service drop-profile (EX8200 Switch)

```
user@switch> show class-of-service drop-profile
Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: dp1, Type: interpolated, Index: 40496
  Fill level    Drop probability
           0                   0
           1                  80
           2                  90
           4                  90
           5                  90
           6                  90
           8                  90
```


Copyright © 2019, Juniper Networks, Inc.

```
          10                  90
          12                  91
          14                  91
          15                  91
          16                  91
          18                  91
          20                  91
          22                  92
          24                  92
          25                  92
          26                  92
          28                  92
          30                  92
          32                  93
          34                  93
          35                  93
          36                  93
          38                  93
          40                  93
          42                  94
          44                  94
          45                  94
          46                  94
          48                  94
          49                  94
          51                  95
          52                  95
          54                  95
          55                  95
          56                  95
          58                  95
          60                  95
          62                  96
          64                  96
          65                  96
          66                  96
          68                  96
          70                  96
          72                  97
          74                  97
          75                  97
          76                  97
          78                  97
          80                  97
          82                  98
          84                  98
          85                  98
          86                  98
          88                  98
          90                  98
          92                  99
          94                  99
          95                  99
          96                  99
          98                  99
          99                  99
         100                 100
Drop profile: dp2, Type: discrete, Index: 40499
  Fill level    Drop probability
```

Copyright © 2019, Juniper Networks, Inc.

10	5
50	50

Copyright © 2019, Juniper Networks, Inc.

show class-of-service fabric scheduler-map

Syntax show class-of-service fabric scheduler-map

Release Information Command introduced before Junos OS Release 7.4.

Description (M320 routers, MX Series routers, T Series routers and EX Series switches only) Display the mapping of class-of-service (CoS) schedulers to switch fabric traffic priorities and a summary of scheduler parameters for each priority.

Options This command has no options.

Required Privilege Level view

List of Sample Output show class-of-service fabric scheduler-map on page 1337

Output Fields Table 151 on page 1337 describes the output fields for the **show class-of-service fabric scheduler-map** command. Output fields are listed in the approximate order in which they appear.

Table 151: show class-of-service fabric scheduler-map Output Fields

Field Name	Field Description
Fabric priority	Indicates the fabric traffic priority. Currently, two priorities are supported: **low** and **high**.
Scheduler	Name of the scheduler.
Index	Index of the indicated object. Objects that have indexes in this output include schedulers and drop profiles.
Drop profiles	Display the assignment of drop profile 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 fabric scheduler-map

```
user@host> show class-of-service fabric scheduler-map
```

```
Fabric priority: low
  Scheduler: fab-ef-scheduler, Index: 60211
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP     44321    fab-ef-profile
```

Copyright © 2019, Juniper Networks, Inc.

```
      Low             TCP         44321    fab-ef-profile
      High            non-TCP     44321    fab-ef-profile
      High            TCP         44321    fab-ef-profile

Fabric priority: high
  Scheduler: fab-ef-scheduler, Index: 60211
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             non-TCP     44321    fab-ef-profile
      Low             TCP         44321    fab-ef-profile
      High            non-TCP     44321    fab-ef-profile
      High            TCP         44321    fab-ef-profile
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service fabric statistics

Syntax

```
show class-of-service fabric statistics
<destination fpc-number>
<detail>
<source fpc-number>
<summary>
```

Release Information Command introduced before Junos OS Release 7.4.

Description (M120, M320, MX240, MX480, MX960, MX2010, MX2020, and T Series routers only) Display class-of-service (CoS) switch fabric queue statistics.

NOTE: On the Switch Control Board (SCB) and the SCBE on MX Series routers, the ratio between high-priority queue and low-priority queue for traffic scheduled to enter the fabric is 85:15. However, on the SCBE2, this ratio is 97:3.

NOTE: After an FPC restart, executing this command can return an **Error = Operation timed out** message for up to a minute even though the FPC is back online. No statistics are lost during this time, however.

Options

none—Same as summary.

destination *fpc-number*—(Optional) Display details for the specified destination Flexible PIC Concentrator (FPC). The FPC number is a value from 0 through 7.

detail—(Optional) Display detailed statistics at the PFE level.

source *fpc-number*—(Optional) Display details for the specified source FPC. The FPC number is a value from 0 through 7.

summary—(Optional) Display all switch fabric statistics.

Required Privilege Level view

List of Sample Output

show class-of-service fabric statistics on page 1340
show class-of-service fabric statistics detail on page 1341

Output Fields Table 152 on page 1340 describes the output fields for the **show class-of-service fabric statistics** command. Output fields are listed in the approximate order in which they appear.

Copyright © 2019, Juniper Networks, Inc.

Table 152: show class-of-service fabric statistics Output Fields

Field Name	Field Description
Destination FPC Index	Index number associated with the destination FPC
Source PFC Index	Index number associated with the source FPC.
Total statistics	Fabric queue statistic totals: • **Packets**—Total packet count for high-priority and low-priority queues. • **Bytes**—Total byte count for high-priority and low-priority queues. • **pps**—Total packets-per-second count for high-priority and low-priority queues. • **bps**—Total bits-per-second count for high-priority and low-priority queues.
Tx statistics	Fabric queue statistics for transmitted traffic: • **Packets**—Transmitted packet count for high-priority and low-priority queues. • **Bytes**—Transmitted byte count for high-priority and low-priority queues. • **pps**—Transmitted packets-per-second count for high-priority and low-priority queues. • **bps**—Transmitted bits-per-second count for high-priority and low-priority queues.
Drop statistics	Fabric queue statistics for dropped traffic, including packets dropped because of internal error: • **Packets**—Dropped packet count for high-priority and low-priority queues. • **Bytes**—Dropped byte count for high-priority and low-priority queues. • **pps**—Dropped packets-per-second count for high-priority and low-priority queues. • **bps**—Dropped bits-per-second count for high-priority and low-priority queues.
Qdepth statistics	Fabric queue depth statistics • **Average**—Average queue depth in bytes. • **Current**—Current queue depth in bytes. • **Max**—Maximum queue depth in bytes.

Sample Output

show class-of-service fabric statistics

```
user@host> show class-of-service fabric statistics
Destination FPC Index: 0, Source FPC Index: 0
 Total statistics:   High priority          Low priority
    Packets:                    0                     0
    Bytes  :                    0                     0
    Pps    :                    0                     0
    bps    :                    0                     0
 Tx statistics:      High priority          Low priority
    Packets:                    0                     0
    Bytes  :                    0                     0
    Pps    :                    0                     0
    bps    :                    0                     0
 Drop statistics:    High priority          Low priority
    Packets:                    0                     0
    Bytes  :                    0                     0
```


Copyright © 2019, Juniper Networks, Inc.

```
   Pps    :                    0                     0
   bps    :                    0                     0

Destination FPC Index: 0, Source FPC Index: 1
 Total statistics:   High priority          Low priority
   Packets:                   0                     0
   Bytes  :                   0                     0
   Pps    :                   0                     0
   bps    :                   0                     0
 Tx statistics:      High priority          Low priority
   Packets:                   0                     0
   Bytes  :                   0                     0
   Pps    :                   0                     0
   bps    :                   0                     0
 Drop statistics:    High priority          Low priority
   Packets:                   0                     0
   Bytes  :                   0                     0
...
```

show class-of-service fabric statistics detail

```
user@host> show class-of-service fabric statistics detail
Destination FPC Index: 4, Destination Pfe Index: 0, Source FPC Index: 4, Source
Pfe Index: 0
 Total statistics:   High priority          Low priority
   Packets:                28953                     0
   Bytes  :             14823936                     0
   Pps    :                   19                     0
   bps    :                81024                     0
 Tx statistics:      High priority          Low priority
   Packets:                28953                     0
   Bytes  :             14823936                     0
   Pps    :                   19                     0
   bps    :                81024                     0
 Drop statistics:    High priority          Low priority
   Packets:                   0                     0
   Bytes  :                   0                     0
   Pps    :                   0                     0
   bps    :                   0                     0
 Qdepth statistics:    High priority           Low priority
   Average:                   0                     0 b
   Current:                   0                     0 b
   Peak   :                   0                     0 b
   Max    :          1367343104            1367343104 b
```

show class-of-service forwarding-table

List of Syntax

Syntax on page 1342
Syntax (TX Matrix and TX Matrix Plus Router) on page 1342

Syntax

```
show class-of-service forwarding-table
```

Syntax (TX Matrix and TX Matrix Plus Router)

```
show class-of-service forwarding-table
<lcc number> | <sfc number>
```

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 14.1X53-D20 for the OCX Series.

Description

Display the entire class-of-service (CoS) configuration as it exists in the forwarding table. Executing this command is equivalent to executing all **show class-of-service forwarding-table** commands in succession.

Options

lcc *number*—(TX Matrix and TX Matrix Plus router only) (Optional) On a TX Matrix router, display the forwarding table configuration for a specific T640 router (or line-card chassis) configured in a routing matrix. On a TX Matrix Plus router, display the forwarding table configuration for a specific router (or line-card chassis) configured in the routing matrix.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

sfc *number*—(TX Matrix Plus routers only) (Optional) Display the forwarding table configuration for the TX Matrix Plus router. Replace ***number*** with **0**.

Required Privilege Level

view

List of Sample Output

show class-of-service forwarding-table on page 1343
show class-of-service forwarding-table lcc (TX Matrix Plus Router) on page 1344

Copyright © 2019, Juniper Networks, Inc.

Output Fields See the output field descriptions for **show class-of-service forwarding-table** commands:

- show class-of-service forwarding-table classifier
- show class-of-service forwarding-table classifier mapping
- show class-of-service forwarding-table drop-profile
- show class-of-service forwarding-table fabric scheduler-map
- show class-of-service forwarding-table rewrite-rule
- show class-of-service forwarding-table rewrite-rule mapping
- show class-of-service forwarding-table scheduler-map

Sample Output

show class-of-service forwarding-table

```
user@host> show class-of-service forwarding-table
Classifier table index: 9, # entries: 8, Table type: EXP
Entry #   Code point   Forwarding-class #   PLP
   0          000        0                 0
   1          001        0                 1
   2          010        1                 0
   3          011        1                 1
   4          100        2                 0
   5          101        2                 1
   6          110        3                 0
   7          111        3                 1

                       Table Index/
Interface      Index      Q num      Table type
sp-0/0/0.1001     66         11      IPv4 precedence
sp-0/0/0.2001     67         11      IPv4 precedence
sp-0/0/0.16383    68         11      IPv4 precedence
fe-0/0/0.0        69         11      IPv4 precedence

Interface: sp-0/0/0 (Index: 129, Map index: 2, Map type: FINAL,
 Num of queues: 2):
  Entry 0 (Scheduler index: 16, Forwarding-class #: 0):
    Tx rate: 0 Kb (95%), Buffer size: 95 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1
  Entry 1 (Scheduler index: 18, Forwarding-class #: 3):
    Tx rate: 0 Kb (5%), Buffer size: 5 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1

Interface: fe-0/0/0 (Index: 137, Map index: 2, Map type: FINAL,
 Num of queues: 2):
  Entry 0 (Scheduler index: 16, Forwarding-class #: 0):
    Tx rate: 0 Kb (95%), Buffer size: 95 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1
  Entry 1 (Scheduler index: 18, Forwarding-class #: 3):
    Tx rate: 0 Kb (5%), Buffer size: 5 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1
```

Copyright © 2019, Juniper Networks, Inc.

```
Interface: fe-0/0/1 (Index: 138, Map index: 2, Map type: FINAL,
 Num of queues: 2):
  Entry 0 (Scheduler index: 16, Forwarding-class #: 0):
    Tx rate: 0 Kb (95%), Buffer size: 95 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1
  Entry 1 (Scheduler index: 18, Forwarding-class #: 3):
    Tx rate: 0 Kb (5%), Buffer size: 5 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1

...

RED drop profile index: 1, # entries: 1
                              Drop
Entry      Fullness(%)    Probability(%)
    0              100              100
```

show class-of-service forwarding-table lcc (TX Matrix Plus Router)

```
user@host> show class-of-service forwarding-table lcc 0
```

```
lcc0-re0:
--------------------------------------------------------------------------

Classifier table index: 9, # entries: 64, Table type: IPv6 DSCP
Entry #   Code point   Forwarding-class #   PLP
   0        000000        0       0
   1        000001        0       0
   2        000010        0       0
   3        000011        0       0
   4        000100        0       0
   5        000101        0       0
   6        000110        0       0
   7        000111        0       0
   8        001000        0       0
   9        001001        0       0
  10        001010        0       0
  11        001011        0       0
  12        001100        0       0
  13        001101        0       0
  14        001110        0       0
  15        001111        0       0
  16        010000        0       0
  17        010001        0       0
  18        010010        0       0
  19        010011        0       0
  20        010100        0       0
  21        010101        0       0
  22        010110        0       0
  23        010111        0       0
  24        011000        0       0
  25        011001        0       0
  26        011010        0       0
  27        011011        0       0
  28        011100        0       0
  29        011101        0       0
  30        011110        0       0
  31        011111        0       0
  32        100000        0       0
```


Copyright © 2019, Juniper Networks, Inc.

```
  33          100001          0          0
  34          100010          0          0
  35          100011          0          0
  36          100100          0          0
  37          100101          0          0
  38          100110          0          0
  39          100111          0          0
  40          101000          0          0
  41          101001          0          0
  42          101010          0          0
  43          101011          0          0
  44          101100          0          0
  45          101101          0          0
  46          101110          0          0
...
```

show class-of-service forwarding-table classifier

Syntax

```
show class-of-service forwarding-table classifier
```

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 14.1X53-D20 for the OCX Series.

Description

Display the mapping of code point value to queue number and loss priority for each classifier as it exists in the forwarding table.

Options

This command has no options.

Required Privilege Level

view

List of Sample Output

show class-of-service forwarding-table classifier on page 1347

Output Fields

Table 153 on page 1346 describes the output fields for the **show class-of-service forwarding-table classifier** command. Output fields are listed in the approximate order in which they appear.

Table 153: show class-of-service forwarding-table classifier Output Fields

Field Name	Field Description
Classifier table index	Index of the classifier table.
entries	Total number of entries.
Table type	Type of code points in the table: **DSCP**, **EXP** (not on the QFX Series), **IEEE 802.1**, **IPv4 precedence** (not on the QFX Series), or **IPv6 DSCP**.
Entry #	Entry number.
Code point	Code point value used for classification.
Forwarding-class #	Forwarding class to which the code point is assigned.
PLP	Packet loss priority value set by classification. For most platforms, the value can be **0** or **1**. For some platforms, the value is **0**, **1**, **2**, or **3**. The value **0** represents low PLP. The value **1** represents **high** PLP. The value **2** represents medium-low PLP. The value **3** represents medium-high PLP.

Copyright © 2019, Juniper Networks, Inc.

Sample Output

show class-of-service forwarding-table classifier

```
user@host> show class-of-service forwarding-table classifier

Classifier table index: 62436, # entries: 64, Table type: DSCP

Entry #   Code point   Forwarding-class #   PLP
   0        000000       0                  0
   1        000001       0                  0
   2        000010       0                  0
   3        000011       0                  0
   4        000100       0                  0
   5        000101       0                  0
   6        000110       0                  0
   7        000111       0                  0
   8        001000       0                  0
   9        001001       0                  0
  10        001010       1                  1
  11        001011       0                  0
...
  60        111100       0                  0
  61        111101       0                  0
  62        111110       0                  0
  63        111111       0                  0
```

show class-of-service forwarding-table classifier mapping

Syntax show class-of-service forwarding-table classifier mapping

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 14.1X53-D20 for the OCX Series.

Description For each logical interface, display either the table index of the classifier for a given code point type or the queue number (if it is a fixed classification) in the forwarding table.

Options This command has no options.

Required Privilege Level view

List of Sample Output show class-of-service forwarding-table classifier mapping on page 1348

Output Fields Table 154 on page 1348 describes the output fields for the **show class-of-service forwarding-table classifier mapping** command. Output fields are listed in the approximate order in which they appear.

Table 154: show class-of-service forwarding-table classifier mapping Output Fields

Field Name	Field Description
Table index/ Q num	If the table type is **Fixed**, the number of the queue to which the interface is mapped. For all other types, this value is the classifier index number.
Interface	Name of the logical interface. This field can also show the physical interface (QFX Series).
Index	Logical interface index.
Table type	Type of code points in the table: **DSCP**, **EXP** (not on the QFX Series), **Fixed**, **IEEE 802.1**, **IPv4 precedence** (not on the QFX Series),or **IPv6 DSCP**. **none** if **no-default** option set.

Sample Output

show class-of-service forwarding-table classifier mapping

```
user@host> show class-of-service forwarding-table classifier mapping
                          Table index/
Interface      Index        Q num      Table type
so-5/0/0.0        10        62436      DSCP
so-0/1/0.0        11        62436      DSCP
```


Copyright © 2019, Juniper Networks, Inc.

```
so-0/2/0.0          12          1      Fixed
so-0/2/1.0          13      62436      DSCP
so-0/2/1.0          13      62437      IEEE 802.1
so-0/2/2.0          14      62436      DSCP
so-0/2/2.0          14      62438      IPv4 precedence
```

Copyright © 2019, Juniper Networks, Inc.

show class-of-service forwarding-table drop-profile

Syntax show class-of-service forwarding-table drop-profile

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 14.1X53-D20 for the OCX Series.

Description Display the data points of all random early detection (RED) drop profiles as they exist in the forwarding table.

Options This command has no options.

Required Privilege Level view

List of Sample Output show class-of-service forwarding-table drop-profile on page 1350

Output Fields Table 155 on page 1350 describes the output fields for the **show class-of-service forwarding-table drop-profile** command. Output fields are listed in the approximate order in which they appear.

Table 155: show class-of-service forwarding-table drop-profile Output Fields

Field Name	Field Description
RED drop profile index	Index of this drop profile.
# entries	Number of entries in a particular RED drop profile index.
Entry	Drop profile entry number.
Fullness(%)	Percentage fullness of a queue.
Drop probability(%)	Drop probability at this fill level.

Sample Output

show class-of-service forwarding-table drop-profile

```
user@host> show class-of-service forwarding-table drop-profile
RED drop profile index: 4, # entries: 1
                                Drop
Entry      Fullness(%)   Probability(%)
    0              100             100

RED drop profile index: 8742, # entries: 3
                                Drop
```


Copyright © 2019, Juniper Networks, Inc.

```
Entry      Fullness(%)   Probability(%)
    0               10              10
    1               20              20
    2               30              30

RED drop profile index: 24627, # entries: 64
                             Drop
Entry      Fullness(%)   Probability(%)
    0                0               0
    1                1               1
    2                2               2
    3                4               4
 ...
   61               98              99
   62               99              99
   63              100             100

RED drop profile index: 25393, # entries: 64
                             Drop
Entry      Fullness(%)   Probability(%)
    0                0               0
    1                1               1
    2                2               2
    3                4               4
 ...
   61               98              98
   62               99              99
   63              100             100
```

show class-of-service forwarding-table fabric scheduler-map

Syntax show class-of-service forwarding-table fabric scheduler-map

Release Information Command introduced before Junos OS Release 7.4.

Description (M320 routers, MX Series routers, T Series routers and EX Series switches only) Display the scheduler map information as it exists in the forwarding table for switch fabric.

Options This command has no options.

Additional Information For information about how packet loss priority is assigned to packets, see "Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows" on page 361.

Required Privilege Level view

List of Sample Output show class-of-service forwarding-table fabric scheduler-map on page 1352

Output Fields Table 156 on page 1352 describes the output fields for the **show class-of-service forwarding-table fabric scheduler-map** command. Output fields are listed in the approximate order in which they appear.

Table 156: show class-of-service forwarding-table fabric scheduler-map Output Fields

Field Name	Field Description
Fabric priority	Fabric traffic priority: **low** and **high**.
Scheduler index	Index of the scheduler applied to a fabric traffic priority.
PLP high	Drop profile index for high-packet-loss-priority (PLP) packets.
PLP low	Drop profile index for low-PLP packets.
TCP PLP high	Drop profile index for low-PLP and Transmission Control Protocol (TCP) packets.
TCP PLP low	Drop profile index for high-PLP and TCP packets.

Sample Output

show class-of-service forwarding-table fabric scheduler-map

```
user@host> show class-of-service forwarding-table fabric scheduler-map
```


Copyright © 2019, Juniper Networks, Inc.

```
Fabric priority: low
  Scheduler index: 60211
    PLP high: 44321, PLP low: 44321, TCP PLP high: 44321, TCP PLP low: 44321

Fabric priority: high
  Scheduler index: 60211
    PLP high: 44321, PLP low: 44321, TCP PLP high: 44321, TCP PLP low: 44321
```

Copyright © 2019, Juniper Networks, Inc.

show class-of-service forwarding-table rewrite-rule

Syntax show class-of-service forwarding-table rewrite-rule

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 14.1X53-D20 for the OCX Series.

Description Display mapping of queue number and loss priority to code point value for each rewrite rule as it exists in the forwarding table.

Options This command has no options.

Required Privilege Level view

List of Sample Output show class-of-service forwarding-table rewrite-rule on page 1354

Output Fields Table 157 on page 1354 describes the output fields for the **show class-of-service forwarding-table rewrite-rule** command. Output fields are listed in the approximate order in which they appear.

Table 157: show class-of-service forwarding-table rewrite-rule Output Fields

Field Name	Field Description
Rewrite table index	Index for this rewrite rule.
# entries	Number of entries in this rewrite rule.
Table type	Type of table: **DSCP**, **EXP** (not on the QFX Series), **EXP-PUSH-3** (not on the QFX Series), **IEEE 802.1**,**IPv4 precedence** (not on the QFX Series), **IPv6 DSCP**, or **Fixed**.
Q#	Queue number to which this entry is assigned.
Low bits	Code point value for low-priority loss profile.
State	State of this code point: **enabled**, **rewritten**, or **disabled**.
High bits	Code point value for high-priority loss profile.

Sample Output

show class-of-service forwarding-table rewrite-rule

```
user@host> show class-of-service forwarding-table rewrite-rule
```


Copyright © 2019, Juniper Networks, Inc.

```
Rewrite table index: 3753, # entries: 4, Table type: DSCP
Q#     Low bits  State      High bits  State
 0      000111  Enabled      001010   Enabled
 2      000000  Disabled     001100   Enabled
 1      101110  Enabled      110111   Enabled
 3      110000  Enabled      111000   Enabled
```

Copyright © 2019, Juniper Networks, Inc.

show class-of-service forwarding-table rewrite-rule mapping

Syntax show class-of-service forwarding-table rewrite-rule mapping

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 14.1X53-D20 for the OCX Series.

Description For each logical interface, display the table identifier of the rewrite rule map for each code point type.

Options This command has no options.

Required Privilege Level view

List of Sample Output show class-of-service forwarding-table rewrite-rule mapping on page 1356

Output Fields Table 158 on page 1356 describes the output fields for the **show class-of-service forwarding-table rewrite-rule mapping** command. Output fields are listed in the approximate order in which they appear.

Table 158: show class-of-service forwarding-table rewrite-rule mapping Output Fields

Field Name	Field Description
Interface	Name of the logical interface. This field can also show the physical interface (QFX Series).
Index	Logical interface index.
Table index	Rewrite table index.
Type	Type of classifier: **DSCP**, **EXP** (not on the QFX Series), **EXP-PUSH-3** (not on the QFX Series), **EXP-SWAP-PUSH-2** (not on the QFX Series),**IEEE 802.1**, **IPv4 precedence** (not on the QFX Series), **IPv6 DSCP**, or **Fixed**.

Sample Output

show class-of-service forwarding-table rewrite-rule mapping

```
user@host> show class-of-service forwarding-table rewrite-rule mapping
Interface      Index   Table index  Type
so-5/0/0.0        10        3753  DSCP
so-0/1/0.0        11        3753  DSCP
so-0/2/0.0        12        3753  DSCP
so-0/2/1.0        13        3753  DSCP
```


Copyright © 2019, Juniper Networks, Inc.

```
so-0/2/2.0          14       3753  DSCP
so-0/2/3.0          15       3753  DSCP
```

Copyright © 2019, Juniper Networks, Inc.

show class-of-service forwarding-table scheduler-map

Syntax show class-of-service forwarding-table scheduler-map

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 14.1X53-D20 for the OCX Series.

Description For each physical interface, display the scheduler map information as it exists in the forwarding table.

Options This command has no options.

Required Privilege Level view

List of Sample Output show class-of-service forwarding-table scheduler-map on page 1359

Output Fields Table 159 on page 1358 describes the output fields for the **show class-of-service forwarding-table scheduler-map** command. Output fields are listed in the approximate order in which they appear.

Table 159: show class-of-service forwarding-table scheduler-map Output Fields

Field Name	Field Description
Interface	Name of the physical interface.
Index	Physical interface index.
Map index	Scheduler map index.
Num of queues	Number of queues defined in this scheduler map.
Entry	Number of this entry in the scheduler map.
Scheduler index	Scheduler policy index.
Forwarding-class #	Forwarding class number to which this entry is applied.
Tx 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.
Max buffer delay	Amount of transmit delay (in milliseconds) or buffer size of the queue. This amount is a percentage of the total interface buffer allocation or the keyword **remainder**, which indicates that the buffer is sized according to what remains after other scheduler buffer allocations.

Copyright © 2019, Juniper Networks, Inc.

Table 159: show class-of-service forwarding-table scheduler-map Output Fields (continued)

Field Name	Field Description
Priority	• **high**—Queue priority is high. • **low**—Queue priority is low.
PLP high	Drop profile index for a high packet loss priority profile.
PLP low	Drop profile index for a low packet loss priority profile.
PLP medium-high	Drop profile index for a medium-high packet loss priority profile.
PLP medium-low	Drop profile index for a medium-low packet loss priority profile.
TCP PLP high	Drop profile index for a high TCP packet loss priority profile.
TCP PLP low	Drop profile index for a low TCP packet loss priority profile.
Policy is exact	If this line appears in the output, exact rate limiting is enabled. Otherwise, no rate limiting is enabled.

Sample Output

show class-of-service forwarding-table scheduler-map

```
user@host> show class-of-service forwarding-table scheduler-map
Interface: so-5/0/0 (Index: 9, Map index: 17638, Num of queues: 2):
  Entry 0 (Scheduler index: 6090, Forwarding-class #: 0):
    Tx rate: 0 Kb (30%), Max buffer delay: 39 bytes (0%)
    Priority low
    PLP high: 25393, PLP low: 24627, TCP PLP high: 25393, TCP PLP low:8742
    Policy is exact
  Entry 1 (Scheduler index: 38372, Forwarding-class #: 1):
    Traffic chunk: Max = 0 bytes, Min = 0 bytes
    Tx rate: 0 Kb (40%), Max buffer delay: 68 bytes (0%)
    Priority high
    PLP high: 25393, PLP low: 24627, TCP PLP high: 25393, TCP PLP low: 8742

Interface: at-6/1/0 (Index: 10, Map index: 17638, Num of queues: 2):
  Entry 0 (Scheduler index: 6090, Forwarding-class #: 0):
    Traffic chunk: Max = 0 bytes, Min = 0 bytes
    Tx rate: 0 Kb (30%), Max buffer delay: 39 bytes (0%)
    Priority high
    PLP high: 25393, PLP low: 24627, TCP PLP high: 25393, TCP PLP low: 8742
  Entry 1 (Scheduler index: 38372, Forwarding-class #: 1):
    Traffic chunk: Max = 0 bytes, Min = 0 bytes
    Tx rate: 0 Kb (40%), Max buffer delay: 68 bytes (0%)
    Priority low
    PLP high: 25393, PLP low: 24627, TCP PLP high: 25393, TCP PLP low: 8742
```

Copyright © 2019, Juniper Networks, Inc.

show class-of-service forwarding-table traffic-class-map

Syntax

```
show class-of-service forwarding-table traffic-class-map <mapping>
```

Release Information

Command introduced in Junos OS Release 14.2 for T4000 routers with Type 5 FPCs.
Command introduced in Junos OS Release 17.2 for MX Routers with MPCs.

Description

Display the mapping of code point value to the traffic class map as it exists in the forwarding table.

Options

mapping—Display the mapping of interfaces to traffic class maps.

Required Privilege Level

view

Related Documentation

- traffic-class-map on page 1302
- Managing Ingress Oversubscription at the PFE on page 660
- Configuring Traffic Class Maps to Manage Ingress Oversubscription on page 661
- Example: Configuring Traffic Class Maps on page 665
- show class-of-service traffic-class-map on page 1409

List of Sample Output

show class-of-service forwarding-table traffic-class-map on page 1361
show class-of-service forwarding-table traffic-class-map mapping on page 1361

Output Fields

Table 160 on page 1360 describes the output fields for the **show class-of-service forwarding-table traffic-class-map** command. Output fields are listed in the approximate order in which they appear.

Table 160: show class-of-service forwarding-table traffic-class-map Output Fields

Field Name	Field Description
Traffic-class-map table index	Index of the traffic class map table.
entries	Total number of entries.
Table type	Type of code points in the table: **DSCP**, **EXP** , **IEEE 802.1**, **IEEE 802.1ad**, or **INET-precedence**
Entry #	Entry number.
Code point	Code point value used for classification.
Traffic-class	Traffic class to which the code point is assigned.

Copyright © 2019, Juniper Networks, Inc.

Table 161 on page 1361 describes the output fields for the **show class-of-service forwarding-table traffic-class-map mapping** command. Output fields are listed in the approximate order in which they appear.

Table 161: show class-of-service forwarding-table traffic-class-map mapping Output Fields

Field Name	Field Description
Interface	Interface to which the traffic class map is assigned.
Index	Internal index of the traffic class map.
Table Index	Index of the traffic class map table.
Table type	Type of code points in the table: **DSCP**, **EXP** , **IEEE 802.1**, **IEEE 802.1ad**, or **INET-precedence**

Sample Output

show class-of-service forwarding-table traffic-class-map

```
user@host> show class-of-service forwarding-table traffic-class-map
Traffic-class-map table index: 44231, # entries: 6, Table type: INET-Precedence
        Entry #   Code point   Traffic-class
           0          000      real-time
           1          001      real-time
           2          010      network-control
           3          011      network-control
           4          100      best-effort
           5          101      best-effort
```

Sample Output

show class-of-service forwarding-table traffic-class-map mapping

```
user@host> show class-of-service forwarding-table traffic-class-map mapping
Interface      Index    Table Index   Table type
        xe-4/0/0         210        44231     INET-Precedence
```

show class-of-service fragmentation-map

Syntax show class-of-service fragmentation-map

Release Information Command introduced in Junos OS Release 7.5.

Description For Multiservices and Services PIC link services IQ interfaces (**lsq**) only, display fragmentation properties for specific forwarding classes.

Options This command has no options.

Required Privilege Level view

List of Sample Output show class-of-service fragmentation-map on page 1362

Output Fields Table 162 on page 1362 describes the output fields for the **show class-of-service fragmentation-map** command. Output fields are listed in the approximate order in which they appear.

Table 162: show class-of-service fragmentation-map Output Fields

Field Name	Field Description
Fragmentation map	Name of the class of service (CoS) fragmentation map.
Index	Index number of the CoS fragmentation map.
Forwarding class	Name of the associated forwarding class.
Fragmentation threshold	Maximum size of each multilink fragment.
No Fragmentation	Packets of this class are not fragmented.
Multilink Class	For multilink multiclass PPP only, the multilink class number corresponding to the forwarding class.

Sample Output

show class-of-service fragmentation-map

```
user@host> show class-of-service fragmentation-map
Fragmentation map: fragmap2, Index: 19801
  Forwarding class: fcDefault
  No Fragmentation

Forwarding class: fcCopper
  Fragmentation threshold: 64, Multilink Class: 1
```


Copyright © 2019, Juniper Networks, Inc.

```
  Forwarding class: fcSilver
  Fragmentation threshold: 100, Multilink Class: 0

  Forwarding class: fcCritical
  Fragmentation threshold: 64, Multilink Class: 0

Fragmentation map: fragmap, Index: 23147
  Forwarding class: fcDefault
  No Fragmentation

  Forwarding class: fcSilver
  Fragmentation threshold: 100

  Forwarding class: fcCritical
  Fragmentation threshold: 100
```

Copyright © 2019, Juniper Networks, Inc.

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 Metro routers.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
Options **detail** and **comprehensive** introduced in Junos OS Release 11.4.
Command introduced in Junos OS Release 15.1R3 on MX Series routers for enhanced subscriber management.

Description

Display the logical and physical interface associations for the classifier, rewrite rules, and scheduler map objects.

NOTE: On routing platforms with dual Routing Engines, running this command on the backup Routing Engine, with or without any of the available options, is not supported and produces the following error message:

error: the class-of-service subsystem is not running

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:

Copyright © 2019, Juniper Networks, Inc.

- 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.

none—Display CoS associations for all physical and logical interfaces.

NOTE: ACX5000 routers do not support classification on logical interfaces and therefore do not show CoS associations for logical interfaces with this command.

Required Privilege Level view

Related Documentation
- *Verifying and Managing Junos OS Enhanced Subscriber Management*

List of Sample Output

show class-of-service interface (Physical) on page 1377
show class-of-service interface (Logical) on page 1377
show class-of-service interface (Gigabit Ethernet) on page 1378
show class-of-service interface (ANCP) on page 1378
show class-of-service interface (PPPoE Interface) on page 1378
show class-of-service interface (DHCP Interface) on page 1378
show class-of-service interface (T4000 Routers with Type 5 FPCs) on page 1379
show class-of-service interface detail on page 1379
show class-of-service interface comprehensive on page 1380
show class-of-service interface (ACX Series Routers) on page 1391
show class-of-service interface (PPPoE Subscriber Interface for Enhanced Subscriber Management) on page 1393

Output Fields Table 163 on page 1365 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 163: 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. (Enhanced subscriber management for MX Series routers) Index values for dynamic CoS traffic control profiles and dynamic scheduler maps are larger for enhanced subscriber management than they are for legacy subscriber management.

Copyright © 2019, Juniper Networks, Inc.

Table 163: show class-of-service interface Output Fields (continued)

Field Name	Field Description
Dedicated Queues	Status of dedicated queues configured on an interface. Supported only on Trio MPC/MIC interfaces on MX Series routers. (Enhanced subscriber management for MX-Series routers) This field is not displayed for enhanced subscriber management.
Maximum usable queues	Number of queues you can configure on the interface.
Maximum usable queues	Maximum number of queues you can use.
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. (Enhanced subscriber management for MX Series routers) This field is not displayed for enhanced subscriber management.
Rewrite Input IEEE Code-point	(QFX3500 switches 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. (Enhanced subscriber management for MX Series routers) The name of the dynamic scheduler map object is associated with a generated UID (for example, **SMAP-1_UID1002**) instead of with a subscriber interface.
Scheduler map forwarding class sets	(QFX Series only) Name of the output fabric 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.
Traffic-control-profile	Name of the associated traffic control profile. (Enhanced subscriber management for MX Series routers) The name of the dynamic traffic control profile object is associated with a generated UID (for example, **TC_PROF_100_199_SERIES_UID1006**) instead of with a subscriber interface.
Classifier	Name and type of classifiers associated with this interface.

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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 and EX4600 switches 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**, **Translation Table** (for IQE PICs only), or **traffic-class-map** (for T4000 routers with Type 5 FPCs).
Name	Name of an object.
Type	Type of an object: **dscp**, **dscp-ipv6**, **exp**, **ieee-802.1**, **ip**, **inet-precedence**, or **ieee-802.1ad** (for traffic class map on T4000 routers with Type 5 FPCs)..
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.

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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.

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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.

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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.
Exclude Overhead Bytes	Exclude the counting of overhead bytes from aggregate queue statistics. • **Disabled**—Default configuration. Includes the counting of overhead bytes in aggregate queue statistics. • **Enabled**—Excludes the counting of overhead bytes from aggregate queue statistics for just the physical interface. • **Enabled for hierarchy**—Excludes the counting of overhead bytes from aggregate queue statistics for the physical interface as well as all child interfaces, including logical interfaces and interface sets.
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.

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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 Maximum usable queues 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

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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)

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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.

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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 Maximum usable queues 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.

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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**.

Copyright © 2019, Juniper Networks, Inc.

Table 163: 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.

Copyright © 2019, Juniper Networks, Inc.

Table 163: show class-of-service interface Output Fields (continued)

Field Name	Field Description

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 appear as **DHCP**, which adjusts the shaping-rate and overhead-accounting class-of-service attributes based on DSL Forum VSA conveyed in DHCP option 82, suboption 9 (Vendor Specific Information). 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).
	  ◦ 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**.
	• **Adjustment overhead bytes**—Number of bytes that the ANCP agent adds to or subtracts from the actual downstream frame overhead before reporting the adjusted values to CoS.
	• **Adjustment target**—Level of shaping-rate adjustment performed: **node** or **queue**.
	• **Adjustment multicast index**—

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
Maximum usable queues: 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
```

Copyright © 2019, Juniper Networks, Inc.

```
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
Maximum usable queues: 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 (ANCP)

```
user@host> show class-of-service interface pp0.1073741842
```

```
  Logical interface: pp0.1073741842, Index: 341
Object                  Name                    Type                    Index
Traffic-control-profile TCP-CVLAN               Output                  12408
Classifier              dscp-ipv6-compatibility dscp-ipv6                   9
Classifier              ipprec-compatibility    ip                         13

    Adjusting application: ancp LS-0
      Adjustment type: absolute
      Configured shaping rate: 4000000
      Adjustment value: 11228000
      Adjustment overhead-accounting mode: Frame Mode
      Adjustment overhead bytes: 50
      Adjustment target: node
```

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 (DHCP Interface)

```
user@host> show class-of-service interface demux0.1
```

```
Logical interface: pp0.1, Index: 85
    Object                  Name                    Type            Index
    Traffic-control-profile tcp-dhcp.o.demux0.1     Output          2726446535
```


Copyright © 2019, Juniper Networks, Inc.

```
   Classifier              ipprec-compatibility   ip            13

   Adjusting application: DHCP
     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
    Maximum usable queues: 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
  Maximum usable queues: 4, Queues in use: 5
  Shaping rate: 50000 bps
  Scheduler map: interface-schedular-map, Index: 58414
  Input shaping rate: 10000 bps
  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
```

Copyright © 2019, Juniper Networks, Inc.

```
    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  Exclude Overhead Bytes: Disabled
  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
```


Copyright © 2019, Juniper Networks, Inc.

```
  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
```


Copyright © 2019, Juniper Networks, Inc.

```
  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
Maximum usable queues: 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
```


Copyright © 2019, Juniper Networks, Inc.

```
  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
```


Copyright © 2019, Juniper Networks, Inc.

```
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
```

Copyright © 2019, Juniper Networks, Inc.

```
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
```


Copyright © 2019, Juniper Networks, Inc.

```
    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
```


Copyright © 2019, Juniper Networks, Inc.

```
    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
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
```


Copyright © 2019, Juniper Networks, Inc.

```
Object                  Name                  Type                  Index
Classifier              d1                    dscp                  11331

Physical interface: ge-0/2/0, Index: 154
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 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
Maximum usable queues: 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 (PPPoE Subscriber Interface for Enhanced Subscriber Management)

```
user@host> show class-of-service interface pp0.3221225474
```

```
  Logical interface: pp0.3221225475, Index: 3221225475
Object                  Name                  Type                  Index
Traffic-control-profile TC_PROF_100_199_SERIES_UID1006 Output      4294967312
Scheduler-map           SMAP-1_UID1002        Output                4294967327
Rewrite-Output          ieee-rewrite          ieee8021p                60432
Rewrite-Output          rule1                 ip                       50463

    Adjusting application: PPPoE IA tags
      Adjustment type: absolute
      Configured shaping rate: 11000000
      Adjustment value: 5000000
      Adjustment target: node

    Adjusting application: ucac
```

```
    Adjustment type: delta
    Configured shaping rate: 5000000
    Adjustment value: 100000
    Adjustment target: node
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service loss-priority-rewrite

Syntax

```
show class-of-service loss-priority-rewrite
<name name>
<type frame-relay-de>
```

Release Information Command introduced in Junos OS Release 11.4.

Description Display the mapping of the code-point value to the loss priority rewrite rule.

Options

none—Display all loss priority rewrite maps.

name—(Optional) Display the specified loss priority rewrite.

frame-relay-de—(Optional) Display the Frame Relay discard eligibility code-point information.

Required Privilege Level view

Related Documentation

- frame-relay-de on page 1146

List of Sample Output show class-of-service loss-priority-rewrite on page 1395

Output Fields This table describes the output fields for the **show class-of-service loss-priority-rewrite** command. Output fields are listed in the approximate order in which they appear.

Table 164: show class-of-service loss-priority-rewrite Output Fields

Field Name	Field Description
Loss-priority-rewrite	Name of the loss priority rewrite.
Code point type	Type: **frame-relay-de**.
Index	Internal index.
Loss priority	Loss priority of **low**, **medium-low**, **medium-high**, or **high**.
Code point	Code-point value.

Sample Output

show class-of-service loss-priority-rewrite

```
user@host> show class-of-service loss-priority-rewrite
```

Copyright © 2019, Juniper Networks, Inc.

```
Loss-priority-rewrite: frame-relay-de-default, Code point type: frame-relay-de,
Index: 38
  Loss priority      Code point
  low                 0
  high                1
  medium-low          0
  medium-high         1
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service l2tp-session

Syntax show class-of-service l2tp-session *session-id*

Release Information Command introduced in Junos OS Release 8.2.

Description Display CoS objects associated with an L2TP session on M7i, M10i, and M120 routers.

Options ***session-id***—L2TP session number for which you want to display a summary of CoS attributes.

Required Privilege Level view

List of Sample Output show class-of-service l2tp-session on page 1398

Output Fields Table 165 on page 1397 lists the output fields for the **show class-of-service l2tp-session** command. Output fields are listed in the approximate order in which they appear.

Table 165: show class-of-service l2tp-session Output Fields

Field Name	Field Description
L2TP Session Username	Username associated with the L2TP session.
Index	Session index identification number.
Physical interface	Physical interface on which the tunnel session is established.
Index	Index ID associated with the physical interface on which the tunnel session is established.
Queues supported	Number of scheduler queues supported for the L2TP session.
Queues in use	Number of scheduler queues active on the L2TP session.
Scheduler map	Scheduler map name associated with the session.
Index	Scheduler map index number associated with the session.
Shaping rate	Maximum bandwidth configured for the session. Each active queue on the session receives a maximum of the configured amount of absolute bandwidth or the configured percentage of bandwidth, even if more bandwidth is available.

Copyright © 2019, Juniper Networks, Inc.

Sample Output

show class-of-service l2tp-session

```
user@host> show class-of-service l2tp-session 123
```

```
L2TP Session Username: user1@example.com, Index: 12553
Physical interface: ge-2/0/0, Index: 130
Queues supported: 8, Queues in use: 4
  Scheduler map: GEN-SCHED-MAP-EF-65%, Index: 5212
  Shaping rate: 200000 bps
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service policy-map

Syntax

```
show class-of-service policy-map
<policy-map-name>
<type (configured | reserved)>
```

Release Information Command introduced in Junos OS Release 16.1.

Description (MPCs on MX Series devices only) Display class-of-service (CoS) policy map information.

Options

policy-map-name—(Optional) Enter the name of a policy map to show only the information for that policy map. Otherwise, information for all policy maps is displayed.

type—(Optional) Display information on different types of policy maps.

configured—Display user configured policy maps.

reserved—Display reserved policy map name-to-ID mapping.

Required Privilege Level view

Related Documentation

- policy-map on page 1231
- Assigning Rewrite Rules on a Per-Customer Basis Using Policy Maps Overview on page 469
- Configuring Policy Maps to Assign Rewrite Rules on a Per-Customer Basis on page 471

List of Sample Output show class-of-service policy-map on page 1400

Output Fields Table 166 on page 1399 describes the output fields for the **show class-of-service policy-map** command. Output fields are listed in the approximate order in which they appear.

Table 166: show class-of-service policy-map Output Fields

Field Name	Field Description
Type	The type of packet marking to rewrite.
Code Point	The code point the packet marking should be rewritten to.
Option	The type of the traffic the packet marking should be rewritten for.

Copyright © 2019, Juniper Networks, Inc.

Sample Output

show class-of-service policy-map

```
user@host> show class-of-service policy-map
Policy-map: P-1, Index: 1
  Type               Code Point  Option
  inet-precedence         110    (proto-ip)
  inet-precedence         110    (proto-mpls)
  dscp-ipv6            101010    (proto-ip)
  dscp-ipv6            101010    (proto-mpls)
  exp                     110    (all-label)
  exp                     111    (outer-label)
  ieee-802.1ad           0110    (outer-and-inner)
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service rewrite-rule

Syntax

```
show class-of-service rewrite-rule
<name name>
<type type>
```

Release Information Command introduced before Junos OS Release 7.4.

Description Display the mapping of forwarding classes and loss priority to code point values.

Options

none—Display all rewrite rules.

name *name*—(Optional) Display the specified rewrite rule.

type *type*—(Optional) Display the rewrite rule of the specified type. The rewrite rule type can be one of the following:

- **dscp**—For IPv4 traffic.
- **dscp-ipv6**—For IPv6 traffic.
- **exp**—For MPLS traffic.
- **frame-relay-de**—(SRX Series only) For Frame Relay traffic.
- **ieee-802.1**—For Layer 2 traffic.
- **inet-precedence**—For IPv4 traffic.

Required Privilege Level view

Related Documentation

- *Rewrite Rules Overview*

List of Sample Output show class-of-service rewrite-rule type dscp on page 1402

Output Fields Table 167 on page 1401 describes the output fields for the **show class-of-service rewrite-rule** command. Output fields are listed in the approximate order in which they appear.

Table 167: show class-of-service rewrite-rule Output Fields

Field Name	Field Description
Rewrite rule	Name of the rewrite rule.
Code point type	Type of rewrite rule: **dscp**, **dscp-ipv6**, **exp**, **frame-relay-de**, or **inet-precedence**.

Copyright © 2019, Juniper Networks, Inc.

Table 167: show class-of-service rewrite-rule Output Fields (continued)

Field Name	Field Description
Forwarding class	Classification of a packet affecting the forwarding, scheduling, and marking policies applied as the packet transits the router or switch.
Index	Internal index for this particular rewrite rule.
Loss priority	Loss priority for rewriting.
Code point	Code point value to rewrite.

Sample Output

show class-of-service rewrite-rule type dscp

```
user@host> show class-of-service rewrite-rule type dscp
Rewrite rule: dscp-default, Code point type: dscp
  Forwarding class                  Loss priority       Code point
  gold                              high                000000
  silver                            low                 110000
  silver                            high                111000
  bronze                            low                 001010
  bronze                            high                001100
  lead                              high                101110

Rewrite rule: abc-dscp-rewrite, Code point type: dscp, Index: 3245
Forwarding class                    Loss priority       Code point
  gold                              low                   000111
  gold                              high                  001010
  silver                            low                   110000
  silver                            high                  111000
  bronze                            high                  001100
  lead                              low                   101110
  lead                              high                  110111
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service routing-instance

Syntax

```
show class-of-service routing-instance
<routing-instance-name>
```

Release Information Command introduced before Junos OS Release 7.4.

Description Display mapping of class of service (CoS) objects to routing instances.

Options ***routing-instance-name***—(Optional) Name of a routing instance.

Required Privilege Level view

List of Sample Output show class-of-service routing-instance on page 1403

Output Fields Table 168 on page 1403 describes the output fields for the **show class-of-service routing-instance** command. Output fields are listed in the approximate order in which they appear.

Table 168: show class-of-service routing-instance Output Fields

Field Name	Field Description
Index	Internal index.
Name	Name of an object.
Object	Category of an object: **Classifier**.
Routing instance	Name of a routing instance.
Type	Type: **exp**.

Sample Output

show class-of-service routing-instance

```
user@host> show class-of-service routing-instance
Routing Instance : vpn1
  Object            Name                 Type                 Index
  Classifier        exp-default          exp                      8

Routing Instance : vpn2
  Object            Name                 Type                 Index
  Classifier        test2                exp                  57507
```

Copyright © 2019, Juniper Networks, Inc.

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.
Support for up to four hierarchy levels added in Junos OS Release 16.1.

Description For MPC/MIC interfaces only, display the scheduler hierarchy as well as the shaping rate, guaranteed rate, priorities, and queue weight information for each forwarding class at each hierarchy level.

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
- hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 1154

List of Sample Output show class-of-service scheduler-hierarchy interface on page 1405

Output Fields Table 169 on page 1404 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 169: show class-of-service scheduler-hierarchy interface Output Fields

Field Name	Field Description
interface	Interface name
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

Copyright © 2019, Juniper Networks, Inc.

Sample Output

show class-of-service scheduler-hierarchy interface

```
user@host> show class-of-service scheduler-hierarchy interface xe-1/0/0

-------------------------------------------------------------------------------
Interface/                    shaping guaranteed  guaranteed/  queue   excess
resource name                    rate      rate       excess  weight   weight
                                kbits     kbits     priority          high/low
-------------------------------------------------------------------------------
xe-1/0/0                        12000
  <<< L1
  xe-1/0/0 RTP                  12000         0                         1    1

    best-effort                 12000         0     Low  Low      950
    network-control             12000         0     Low  Low       50
  ifset1                        12000         0                      500  500
  <<< L2
    ifset1 RTP                  12000         0                         1    1
      be1                         720         0     Low  Low      250
      nc1                       12000         0     Low  Low      250
    demux0.96                    3000         0                         1    1
  <<< L3
      demux0.96 RTP              3000         0                      500  500
        be1                      1000         0     Low  Low      250
        nc1                      3000         0     Low  Low      250
      pp0.81                     2000         0                         1    1
  <<< L4
         be1                     1000         0     Low  Low      250
         nc1                     2000         0     Low  Low      250
```

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.
Command introduced in Junos OS Release 15.1R3 on MX Series routers for enhanced subscriber management.

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

Related Documentation

- *Verifying and Managing Junos OS Enhanced Subscriber Management*

List of Sample Output

show class-of-service scheduler-map on page 1407
show class-of-service scheduler-map (QFX Series) on page 1408

Output Fields

Table 170 on page 1406 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 170: show class-of-service scheduler-map Output Fields

Field Name	Field Description
Scheduler map	Name of the scheduler map. (Enhanced subscriber management for MX Series routers) The name of the dynamic scheduler map object is associated with a generated UID (for example, **SMAP-1_UID1002**) instead of with a subscriber interface.
Index	Index of the indicated object. Objects having indexes in this output include scheduler maps, schedulers, and drop profiles. (Enhanced subscriber management for MX Series routers) Index values for dynamic CoS traffic control profiles are larger for enhanced subscriber management than they are for legacy subscriber management.
Scheduler	Name of the scheduler.

Copyright © 2019, Juniper Networks, Inc.

Table 170: show class-of-service scheduler-map Output Fields (continued)

Field Name	Field Description
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**.
Excess priority	Priority of excess bandwidth: **low**, **medium-low**, **medium-high**, **high**, or **none**.
Explicit Congestion Notification	(QFX Series, OCX Series, and EX4600 switches only) Explicit congestion notification (ECN) state: • Disable—ECN is disabled on the specified scheduler • Enable—ECN is enabled on the specified scheduler ECN is disabled by default.
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:
```

Copyright © 2019, Juniper Networks, Inc.

```
   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 scheduler-map (QFX Series)

```
user@switch# show class-of-service scheduler-map
Scheduler map: be-map, Index: 12240

  Scheduler:be-sched, Forwarding class: best-effort, Index: 115
    Transmit rate: 30 percent, Rate Limit: none, Buffer size: remainder,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified, Explicit Congestion Notification: disable

    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any          3312    lan-dp
      Medium-high     any          2714    be-dp1
      High            any          3178    be-dp2
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service traffic-class-map

Syntax

```
show class-of-service traffic-class-map
<name traffic-class-map-name>
<type (dscp | exp | ieee-802.1 | ieee-802.1ad |inet-precedence)>
```

Release Information

Command introduced in Junos OS Release 14.2 for T4000 routers with Type 5 FPCs.
Command introduced in Junos OS Release 17.2 for MX Routers with MPCs.

Description

For each traffic class map, display the mapping of the code point value to the input traffic class.

Options

none—Display all the mappings.

name *name*—(Optional) Display the named traffic class map.

type dscp—(Optional) Display all traffic class maps of the Differentiated Services code point (DSCP) type.

type exp—(Optional) Display all traffic class maps of the MPLS EXP type.

type ieee-802.1—(Optional) Display all traffic class maps of the IEEE 802.1 type.

type ieee-802.1ad—(Optional) Display all traffic class maps of the IEEE 802.1ad type.

type inet-precedence—(Optional) Display all traffic class maps of the IPv4 precedence type.

Required Privilege Level

view

Related Documentation

- traffic-class-map on page 1302
- Managing Ingress Oversubscription at the PFE on page 660
- Configuring Traffic Class Maps to Manage Ingress Oversubscription on page 661
- Example: Configuring Traffic Class Maps on page 665
- show class-of-service forwarding-table traffic-class-map on page 1360

List of Sample Output

show class-of-service traffic-class-map on page 1410

Output Fields

Table 148 on page 1328 describes the output fields for the **show class-of-service traffic-class-map** command. Output fields are listed in the approximate order in which they appear.

Copyright © 2019, Juniper Networks, Inc.

Table 171: show class-of-service traffic-class-map Output Fields

Field Name	Field Description
Traffic-class-map	Name of the traffic class map.
Code point type	Type of the traffic class map: **exp**, **dscp**, **ieee-802.1**, **ieee-802.1ad**, or **inet-precedence**.
Index	Internal index of the traffic class map.
Code point	Code point value used for classification.
Traffic class	Classification of a packet affecting the forwarding, scheduling, and marking policies applied as the packet transits the router.

Sample Output

show class-of-service traffic-class-map

```
user@host>  show class-of-service traffic-class-map
Traffic-class-map: inet-precedence, Code-point type: inet-precedence, Index: 44231

         Code point         Traffic class
         000                real-time
         001                real-time
         010                network-control
         011                network-control
         100                best-effort
         101                best-effort
```


Copyright © 2019, Juniper Networks, Inc.

show class-of-service translation-table

Syntax

```
show class-of-service translation–table
<name translation-table-name> |
<type (to-dscp–from–dscp | to-dscp-ipv6–from–dscp–ipv6 | to-exp–from–exp |
  to-inet-precedence-from-inet-precedence)>
```

Release Information Command introduced in Junos OS Release 9.3 for IQE PICs.

Description Display the mapping of class-of-service (CoS) translation table code points to corresponding bit patterns.

Options **none**—Display translation table code points for all translation tables.

name—(Optional) Display information for the named translation table.

type—(Optional) Display information for a certain translation table type:

- **to-dscp-from-dscp**—Display DSCP translation table information.
- **to-dscp-ipv6-from-dscp-ipv6**—Display DSCP IPv6 translation table information.
- **to-exp-from-exp**—Display MPLS EXP translation table information.
- **to-inet-precedence-from-intet-precedence**—Display Internet precedence translation table information.

Required Privilege Level view

List of Sample Output

show class-of-service translation-table on page 1413
show class-of-service translation-table name exp-trans-table on page 1414
show class-of-service translation-table type to-dscp-ipv6-from-dscp-ipv6 on page 1414

Output Fields Table 172 on page 1411 describes the output fields for the **show class-of-service translation-table** command. Output fields are listed in the approximate order in which they appear.

Table 172: show class-of-service translation-table Output Fields

Field Name	Field Description
Translation Table	Name of the translation table.
Translation table type	Type of the translation table.
Index	Internal index number of the translation table.
From Code Point	Value of code point received.

Copyright © 2019, Juniper Networks, Inc.

Table 172: show class-of-service translation-table Output Fields (continued)

Field Name	Field Description
To Code Point	Value of translated code point.

Copyright © 2019, Juniper Networks, Inc.

Sample Output

show class-of-service translation-table

```
user@host> show class-of-service translation-table

Translation Table: inet-trans-table, Translation table type: inet-to-inet, Index:
 61075
  From Code point    To Code Point
  000                          101
  001                          111
  010                          101
  011                          111
  100                          101
  101                          101
  110                          001
  111                          000

Translation Table: dscp-trans-table, Translation table type: dscp-to-dscp, Index:
 6761
  From Code point    To Code Point
  000000                    000111
  000001                    000111
  000010                    000111
  000011                    000111
  000100                    000111
  000101                    000111
  000110                    000111
  000111                    111000
  001000                    000111
  001001                    000111
  001010                    000111
  001011                    000111
  001100                    000111
  001101                    000111
  001110                    000111
  001111                    000111
  010000                    000111
  010001                    000111
  010010                    000111
  010011                    000111
  010100                    000111
  010101                    000111
  010110                    000111
  010111                    000111
  011000                    000111
  011001                    000111
  011010                    000111
  011011                    000111
  011100                    000111
  011101                    000111
  011110                    000111
  011111                    000111
  100000                    000111
  100001                    000111
  100010                    000111
  100011                    000111
  100100                    000111
  100101                    000111
  100110                    000111
```

```
  100111                        111000
  101000                        000111
  101001                        000111
  101010                        000111
  101011                        000111
  101100                        000111
  101101                        000111
  101110                        000111
  101111                        000111
  110000                        000111
  110001                        000111
  110010                        000111
  110011                        000111
  110100                        000111
  110101                        000111
  110110                        000111
  110111                        000111
  111000                        000111
  111001                        000111
  111010                        000111
  111011                        000111
  111100                        000111
  111101                        000111
  111110                        000001
  111111                        000000
```

show class-of-service translation-table name exp-trans-table

```
user@host> show class-of-service translation-table name exp-trans-table
Translation Table: exp-trans-table, Translation table type: exp-to-exp, Index:
9048
  From Code point    To Code Point
  000                           101
  001                           111
  010                           101
  011                           111
  100                           101
  101                           101
  110                           001
  111                           000
```

show class-of-service translation-table type to-dscp-ipv6-from-dscp-ipv6

```
user@host> show class-of-service translation-table type to-dscp-ipv6-from-dscp-ipv6
Translation Table: dscp-ipv6-trans-table, Translation table type:
dscp-ipv6-to-dscp-ipv6, Index: 64704
  From Code point    To Code Point
  000000                        000111
  000001                        000111
  000010                        000111
  000011                        000111
  000100                        000111
  000101                        000111
  000110                        000111
  000111                        111000
  001000                        000111
```


Copyright © 2019, Juniper Networks, Inc.

001001	000111
001010	000111
001011	000111
001100	000111
001101	000111
001110	000111
001111	000111
010000	000111
010001	000111
010010	000111
010011	000111
010100	000111
010101	000111
010110	000111
010111	000111
011000	000111
011001	000111
011010	000111
011011	000111
011100	000111
011101	000111
011110	000111
011111	000111
100000	000111
100001	000111
100010	000111
100011	000111
100100	000111
100101	000111
100110	000111
100111	111000
101000	000111
101001	000111
101010	000111
101011	000111
101100	000111
101101	000111
101110	000111
101111	000111
110000	000111
110001	000111
110010	000111
110011	000111
110100	000111
110101	000111
110110	000111
110111	000111
111000	000111
111001	000111
111010	000111
111011	000111
111100	000111
111101	000111
111110	000001
111111	000000

Copyright © 2019, Juniper Networks, Inc.

show interfaces forwarding-class-counters

Syntax show interfaces forwarding-class-counters *interface-name* <comprehensive>

Release Information Command introduced in Junos OS 14.1 for MX Series routers.

Description Display interface accounting information by forwarding class for IPv4, IPv6, MPLS, Layer 2, and Other traffic.

Options

comprehensive—(Optional) Display forwarding-class-counters per traffic family for all logical interfaces under the physical interface along with other quality-of-service information.

Additional Information

For physical interface-level statistics, if none of the logical interfaces have any of the traffic families configured on them, the forwarding class statistics for that family are still displayed with a value of 0.

For physical interface-level statistics, in case of Layer 2 families such as **ccc**, **tcc**, or **vpls**, the **Layer2** keyword is displayed because it is possible that different Layer 2 families are configured on the logical interface.

For logical interface-level statistics, the output displays statistics only for families that are configured on that logical interface. The statistics under **Other** family are still displayed because these are packets that are not classified as belonging to any family.

In the case of Layer 2 families such as **ccc**, **tcc**, or **vpls** configured on the logical interface, the actual family name is displayed in the output.

Statistics include input and output byte and packets and corresponding rates.

Required Privilege Level view

Related Documentation

- forwarding-class-accounting on page 1136
- *Class of Service Feature Guide (Routers and EX9200 Switches)*

List of Sample Output show interfaces forwarding-class-counters interface-name on page 1417

Output Fields Table 173 on page 1417 lists the output fields for the **show interfaces forwarding-class-counters** command. Output fields are listed in the approximate order in which they appear.

Copyright © 2019, Juniper Networks, Inc.

Table 173: show interfaces forwarding-class-counters Output Fields

Field Name	Field Description
Input bytes	A count of received bytes that match the forwarding class.
Output bytes	A count of transmitted bytes that match the forwarding class.
Input packets	A count of received packets that match the forwarding class.
Output packets	A count of transmitted packets that match the forwarding class.

Sample Output

show interfaces forwarding-class-counters interface-name

```
user@host> show interfaces forwarding-class-counters ge-4/2/1
```

```
user@host> show interfaces forwarding-class-counters ge-4/2/1
Physical interface ge-4/2/1 (Index 228) (SNMP ifIndex 870)
    Aggregate Forwarding-class statistics :
       Forwarding-class statistics : best-effort
        Input   bytes   :                    0  0 bps
        output  bytes   :                    0  0 bps
        Input   packets :                    0  0 pps
        output  packets :                    0  0 pps
       Forwarding-class statistics : network-control
        Input   bytes   :                    0  0 bps
        output  bytes   :                    0  0 bps
        Input   packets :                    0  0 pps
        output  packets :                    0  0 pps

    IPv4 Forwarding-class statistics :
       Forwarding-class statistics : best-effort
        Input   bytes   :                    0  0 bps
        output  bytes   :                    0  0 bps
        Input   packets :                    0  0 pps
        output  packets :                    0  0 pps
       Forwarding-class statistics : expedited-forwarding
        Input   bytes   :                    0  0 bps
        output  bytes   :                    0  0 bps
        Input   packets :                    0  0 pps
        output  packets :                    0  0 pps
       Forwarding-class statistics : assured-forwarding
        Input   bytes   :                    0  0 bps
        output  bytes   :                    0  0 bps
        Input   packets :                    0  0 pps
        output  packets :                    0  0 pps
       Forwarding-class statistics : network-control
        Input   bytes   :                    0  0 bps
        output  bytes   :                    0  0 bps
        Input   packets :                    0  0 pps
        output  packets :                    0  0 pps

   IPv6 Forwarding-class statistics :
       Forwarding-class statistics : best-effort
        Input   bytes   :                    0  0 bps
```

Copyright © 2019, Juniper Networks, Inc.

```
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps
    Forwarding-class statistics : expedited-forwarding
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps
    Forwarding-class statistics : assured-forwarding
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps
    Forwarding-class statistics : network-control
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps

MPLS Forwarding-class statistics :
    Forwarding-class statistics : best-effort
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps
    Forwarding-class statistics : expedited-forwarding
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps
    Forwarding-class statistics : assured-forwarding
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps
    Forwarding-class statistics : network-control
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps

Layer2 Forwarding-class statistics
    Forwarding-class statistics : best-effort
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps
    Forwarding-class statistics : expedited-forwarding
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps
    Forwarding-class statistics : assured-forwarding
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
     Input   packets :                    0  0 pps
     output  packets :                    0  0 pps
    Forwarding-class statistics : network-control
     Input   bytes   :                    0  0 bps
     output  bytes   :                    0  0 bps
```


Copyright © 2019, Juniper Networks, Inc.

```
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps

 Other Forwarding-class statistics :
     Forwarding-class statistics : best-effort
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : expedited-forwarding
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : assured-forwarding
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : network-control
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps


Logical interface ge-4/2/1.0 (Index 347) (SNMP ifIndex 1032)
   Forwarding-class accounting parameters :
 Aggregate Forwarding-class statistics :
            Forwarding-class statistics : best-effort
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : expedited-forwarding
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : assured-forwarding
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : network-control
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps

 ccc Forwarding-class statistics :
     Forwarding-class statistics : best-effort
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : expedited-forwarding
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
```

```
      output  packets :                    0  0 pps
     Forwarding-class statistics : assured-forwarding
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : network-control
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps

  Other Forwarding-class statistics :
     Forwarding-class statistics : best-effort
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : expedited-forwarding
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : assured-forwarding
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
     Forwarding-class statistics : network-control
      Input   bytes   :                    0  0 bps
      output  bytes   :                    0  0 bps
      Input   packets :                    0  0 pps
      output  packets :                    0  0 pps
```


Copyright © 2019, Juniper Networks, Inc.

show interfaces queue

Syntax

```
show interfaces queue
<aggregate | remaining-traffic>
<both-ingress-egress>
<egress>
<forwarding-class forwarding-class>
<ingress>
< interface-name>
<l2-statistics>
<buffer-occupancy>
```

Release Information

Command introduced before Junos OS Release 7.4.
both-ingress-egress, **egress**, and **ingress** options introduced in Junos OS Release 7.6.
Command introduced in Junos OS Release 9.0 for EX Series.
Command introduced in Junos OS Release 11.1 for the QFX Series.
l2-statistics option introduced in Junos OS Release 12.1.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
buffer-occupancy statement introduced in Junos OS Release 19.1R1 for QFX5000 Series switches.

Description

Display class-of-service (CoS) queue information for physical interfaces.

Options

none—Show detailed CoS queue statistics for all physical interfaces.

aggregate—(Optional) Display the aggregated queuing statistics of all logical interfaces that have traffic-control profiles configured. (Not on the QFX Series.)

both-ingress-egress—(Optional) On Gigabit Ethernet Intelligent Queuing 2 (IQ2) PICs, display both ingress and egress queue statistics. (Not on the QFX Series.)

egress—(Optional) Display egress queue statistics.

forwarding-class *forwarding-class*—(Optional) Forwarding class name for this queue. Shows detailed CoS statistics for the queue associated with the specified forwarding class.

ingress—(Optional) On Gigabit Ethernet IQ2 PICs, display ingress queue statistics. (Not on the QFX Series.)

interface-name—(Optional) Show detailed CoS queue statistics for the specified interface.

l2-statistics—(Optional) Display Layer 2 statistics for MLPPP, FRF.15, and FRF.16 bundles

buffer-occupancy—Displays the peak buffer occupancy for each queue while **buffer-monitor-enable** is enabled at the **[edit chassis fpc *slot-number* traffic-manager]** hierarchy level.

remaining-traffic—(Optional) Display the remaining-traffic queue statistics of all logical interfaces that have traffic-control profiles configured.

Copyright © 2019, Juniper Networks, Inc.

Overhead for Layer 2 Statistics

Transmitted packets and transmitted byte counts are displayed for the Layer 2 level with the addition of encapsulation overheads applied for fragmentation, as shown in Table 174 on page 1422. Others counters, such as packets and bytes queued (input) and drop counters, are displayed at the Layer 3 level. In the case of link fragmentation and interleaving (LFI) for which fragmentation is not applied, corresponding Layer 2 overheads are added, as shown in Table 174 on page 1422.

Table 174: Layer 2 Overhead and Transmitted Packets or Byte Counts

Protocol	Fragmentation		LFI
	First fragmentation	Second to *n* fragmentations	
	Bytes	Bytes	
MLPPP (Long)	13	12	8
MLPPP (short)	11	10	8
MLFR (FRF15)	12	10	8
MFR (FRF16)	10	8	-
MCMLPPP(Long)	13	12	-
MCMLPPP(Short)	11	10	-

Layer 2 Statistics—Fragmentation Overhead Calculation

```
MLPPP/MC-MLPPP Overhead details:
   ==============================
   Fragment 1:

    Outer PPP header                          : 4 bytes
    Long or short sequence MLPPP header       : 4 bytes or 2 bytes
    Inner PPP header                          : 1 byte
    HDLC flag and FCS bytes                   : 4 bytes

   Fragments 2 .. n :

    Outer PPP header                          : 4 bytes
    Long or short sequence MLPPP header       : 4 bytes or 2 bytes
    HDLC flag and FCS bytes                   : 4 bytes


   MLFR (FRF15) Overhead details:
   ============================
   Fragment 1:
       Framerelay header     :  2 bytes
       Control,NLPID         :  2 bytes
```


Copyright © 2019, Juniper Networks, Inc.

```
      Fragmentaion header   :  2 bytes
      Inner proto           :  2 bytes
      HDLC flag and  FCS    :  4 bytes

  Fragments 2 ...n :

      Framerelay header     :  2 bytes
      Control,NLPID         :  2 bytes
      Fragmentaion header   :  2 bytes
      HDLC flag and  FCS    :  4 bytes


  MFR (FRF16) Overhead details:
  ==============================
  Fragment 1:
     Fragmentaion header   :  2 bytes
     Framerelay header     :  2 bytes
     Inner proto           :  2 bytes
     HDLC flag and  FCS    :  4 bytes

  Fragments 2 ...n :
     Fragmentaion header   :  2 bytes
     Framerelay header     :  2 bytes
     HDLC flag and  FCS    :  4 bytes
```

Overhead with LFI

```
MLPPP(Long & short sequence):
   ============================
      Outer PPP header      :  4 bytes
      HDLC flag and  FCS    :  4 bytes

  MLFR (FRF15):
  =============
      Framerelay header     :  2 bytes
      Control,NLPID         :  2 bytes
      HDLC flag and  FCS    :  4 bytes
```

The following examples show overhead for different cases:

- A 1000-byte packet is sent to a mlppp bundle without any fragmentation. At the Layer 2 level, bytes transmitted is 1013 in 1 packet. This overhead is for MLPPP long sequence encap.
- A 1000-byte packet is sent to a mlppp bundle with a fragment threshold of 250byte. At the Layer 2 level, bytes transmitted is 1061 bytes in 5 packets.
- A 1000-byte LFI packet is sent to an mlppp bundle. At the Layer 2 level, bytes transmitted is 1008 in 1 packet.

remaining-traffic—(Optional) Display the queuing statistics of all logical interfaces that do not have traffic-control profiles configured. (Not on the QFX Series.)

Copyright © 2019, Juniper Networks, Inc.

Additional Information

For rate-limited interfaces hosted on Modular Interface Cards (MICs), Modular Port Concentrators (MPCs), or Enhanced Queuing DPCs, rate-limit packet-drop operations occur *before* packets are queued for transmission scheduling. For such interfaces, the statistics for queued traffic do not include the packets that have already been dropped due to rate limiting, and consequently the displayed statistics for queued traffic are the same as the displayed statistics for transmitted traffic.

NOTE: For rate-limited interfaces hosted on other types of hardware, rate-limit packet-drop operations occur *after* packets are queued for transmission scheduling. For these other interface types, the statistics for queued traffic include the packets that are later dropped due to rate limiting, and consequently the displayed statistics for queued traffic equals the sum of the statistics for transmitted and rate-limited traffic.

On M Series routers (except for the M320 and M120 routers), this command is valid only for a PIC installed on an enhanced Flexible PIC Concentrator (FPC).

Queue statistics for aggregated interfaces are supported on the M Series and T Series routers only. Statistics for an aggregated interface are the summation of the queue statistics of the child links of that aggregated interface. You can view the statistics for a child interface by using the **show interfaces statistics** command for that child interface.

When you configure tricolor marking on a 10-port 1-Gigabit Ethernet PIC, for queues 6 and 7 only, the output does not display the number of queued bytes and packets, or the number of bytes and packets dropped because of RED. If you do not configure tricolor marking on the interface, these statistics are available for all queues.

For the 4-port Channelized OC12 IQE PIC and 1-port Channelized OC48 IQE PIC, the **Packet Forwarding Engine Chassis Queues** field represents traffic bound for a particular physical interface on the PIC. For all other PICs, the **Packet Forwarding Engine Chassis Queues** field represents the total traffic bound for the PIC.

For Gigabit Ethernet IQ2 PICs, the **show interfaces queue** command output does not display the number of tail-dropped packets. This limitation does not apply to Packet Forwarding Engine chassis queues.

When fragmentation occurs on the egress interface, the first set of packet counters shows the postfragmentation values. The second set of packet counters (under the **Packet Forwarding Engine Chassis Queues** field) shows the prefragmentation values.

The behavior of the **egress** queues for the **Routing Engine-Generated Traffic** is not same as the configured queue for MLPPP and MFR configurations.

For related CoS operational mode commands, see the CLI Explorer.

Required Privilege Level

view

Copyright © 2019, Juniper Networks, Inc.

Release History Table

Release	Description
18.3R1	Starting with Junos OS 18.3R1, the **Tail-dropped packets** counter is supported on PTX Series Packet Transport Routers.
16.1	Starting with Junos OS Release 16.1, **Last-packet enqueued** output field is introduced.

Related Documentation

- *Monitoring Interface Status and Traffic*
- *Monitoring Interfaces That Have CoS Components*
- *Defining CoS Schedulers and Scheduler Maps (CLI Procedure)*
- *Configuring CoS Traffic Classification for Ingress Queuing on Oversubscribed Ports on EX8200 Line Cards (CLI Procedure)*
- *ATM Interfaces Feature Guide for Routing Devices*
- *Interfaces Feature Guide*
- *Layer 2 Bridging, Address Learning, and Forwarding Feature Guide*

List of Sample Output

show interfaces queue (Rate-Limited Interface on a Gigabit Ethernet MIC in an MPC) on page 1431
show interfaces queue (Aggregated Ethernet on a T320 Router) on page 1432
show interfaces queue (Gigabit Ethernet on a T640 Router) on page 1433
show interfaces queue aggregate (Gigabit Ethernet Enhanced DPC) on page 1434
show interfaces queue (Gigabit Ethernet IQ2 PIC) on page 1438
show interfaces queue both-ingress-egress (Gigabit Ethernet IQ2 PIC) on page 1441
show interfaces queue ingress (Gigabit Ethernet IQ2 PIC) on page 1443
show interfaces queue egress (Gigabit Ethernet IQ2 PIC) on page 1444
show interfaces queue remaining-traffic (Gigabit Ethernet Enhanced DPC) on page 1446
show interfaces queue (Channelized OC12 IQE Type 3 PIC in SONET Mode) on page 1448
show interfaces queue (QFX Series) on page 1458
show interfaces queue l2-statistics (lsq interface) on page 1459
show interfaces queue lsq (lsq-ifd) on page 1460
show interfaces queue (Aggregated Ethernet on a MX series Router) on page 1462
show interfaces queue ge-0/0/0 (EX2200 Switch) on page 1463
show interfaces queue xe-6/0/39 (Line Card with Oversubscribed Ports in an EX8200 Switch) on page 1464
show interfaces queue xe-0/0/2 buffer-occupancy (QFX5000 Series switch) on page 1467

Output Fields

Table 175 on page 1426 lists the output fields for the **show interfaces queue** command. Output fields are listed in the approximate order in which they appear.

Copyright © 2019, Juniper Networks, Inc.

Table 175: show interfaces queue Output Fields

Field Name	Field Description
Physical interface	Name of the physical interface.
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.
Interface index	Physical interface's index number, which reflects its initialization sequence.
SNMP ifIndex	SNMP index number for the interface.
Forwarding classes supported	Total number of forwarding classes supported on the specified interface.
Forwarding classes in use	Total number of forwarding classes in use on the specified interface.
Ingress queues supported	On Gigabit Ethernet IQ2 PICs only, total number of ingress queues supported on the specified interface.
Ingress queues in use	On Gigabit Ethernet IQ2 PICs only, total number of ingress queues in use on the specified interface.
Output queues supported	Total number of output queues supported on the specified interface.
Output queues in use	Total number of output queues in use on the specified interface.
Egress queues supported	Total number of egress queues supported on the specified interface.
Egress queues in use	Total number of egress queues in use on the specified interface.
Queue counters (Ingress)	CoS queue number and its associated user-configured forwarding class name. Displayed on IQ2 interfaces. • **Queued packets**—Number of queued packets. NOTE: This field is not supported on QFX5100, QFX5110, QFX5200, and QFX5210 switches due to hardware limitations. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism.
Burst size	(Logical interfaces on IQ PICs only) Maximum number of bytes up to which the logical interface can burst. The burst size is based on the shaping rate applied to the interface.
The following output fields are applicable to both interface component and Packet Forwarding component in the **show interfaces queue** command:	
Queue	Queue number.
Forwarding classes	Forwarding class name.

Copyright © 2019, Juniper Networks, Inc.

Table 175: show interfaces queue Output Fields (continued)

Field Name	Field Description
Queued Packets	Number of packets queued to this queue. NOTE: For Gigabit Ethernet IQ2 interfaces, the Queued Packets count is calculated by the Junos OS interpreting one frame buffer as one packet. If the queued packets are very large or very small, the calculation might not be completely accurate for transit traffic. The count is completely accurate for traffic terminated on the router. For rate-limited interfaces hosted on MICs or MPCs only, this statistic does not include traffic dropped due to rate limiting. For more information, see "Additional Information" on page 1424. NOTE: This field is not supported on QFX5100, QFX5110, QFX5200, and QFX5210 switches due to hardware limitations. This field is not supported on EX Series switches due to hardware limitations.
Queued Bytes	Number of bytes queued to this queue. The byte counts vary by interface hardware. For more information, see Table 176 on page 1429. For rate-limited interfaces hosted on MICs or MPCs only, this statistic does not include traffic dropped due to rate limiting. For more information, see "Additional Information" on page 1424. NOTE: This field is not supported on QFX5100, QFX5110, QFX5200, and QFX5210 switches due to hardware limitations. This field is not supported on EX Series switches due to hardware limitations.
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. NOTE: For Layer 2 statistics, see "Overhead for Layer 2 Statistics" on page 1422
Transmitted Bytes	Number of bytes transmitted by this queue. The byte counts vary by interface hardware. For more information, see Table 176 on page 1429. NOTE: On MX Series routers, this number can be inaccurate when you issue the command for a physical interface repeatedly and in quick succession, because the statistics for the child nodes are collected infrequently. Wait ten seconds between successive iterations to avoid this situation. NOTE: For Layer 2 statistics, see "Overhead for Layer 2 Statistics" on page 1422
Tail-dropped packets	Number of packets dropped because of tail drop. NOTE: Starting with Junos OS 18.3R1, the **Tail-dropped packets** counter is supported on PTX Series Packet Transport Routers.
RL-dropped packets	Number of packets dropped due to rate limiting. For rate-limited interfaces hosted on MICs, MPCs, and Enhanced Queuing DPCs only, this statistic is not included in the queued traffic statistics. For more information, see "Additional Information" on page 1424. NOTE: The **RL-dropped packets** counter is not supported on the PTX Series Packet Transport Routers, and is omitted from the output.

Copyright © 2019, Juniper Networks, Inc.

Table 175: show interfaces queue Output Fields (continued)

Field Name	Field Description
RL-dropped bytes	Number of bytes dropped due to rate limiting. For rate-limited interfaces hosted on MICs, MPCs, and Enhanced Queuing DPCs only, this statistic is not included in the queued traffic statistics. For more information, see "Additional Information" on page 1424.

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 interface hardware. For more information, see Table 176 on page 1429.
	
	• (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.

Queue-depth bytes	Displays queue-depth average, current, peak, and maximum values for RTP queues. Because queue-depth values cannot be aggregated, displays the values for RTP queues regardless of whether **aggregate**, **remaining-traffic**, or neither option is selected.
Peak	(QFX5000 Series switches only) Diplays the peak buffer occupancy for the queue while **buffer-monitor-enable** is enabled at the **[edit chassis fpc *slot-number* traffic-manager]** hierarchy level.

Copyright © 2019, Juniper Networks, Inc.

Table 175: show interfaces queue Output Fields (continued)

Field Name	Field Description
Last-packet enqueued	Starting with Junos OS Release 16.1, **Last-packet enqueued** output field is introduced. If **packet-timestamp** is enabled for an FPC, shows the day, date, time, and year in the format *day-of-the-week month day-date hh:mm:ss yyyy* when a packet was enqueued in the CoS queue. When the timestamp is aggregated across all active Packet Forwarding Engines, the latest timestamp for each CoS queue is reported.

Byte counts vary by interface hardware. Table 176 on page 1429 shows how the byte counts on the outbound interfaces vary depending on the interface hardware. Table 176 on page 1429 is based on the assumption that outbound interfaces are sending IP traffic with 478 bytes per packet.

Table 176: Byte Count by Interface Hardware

Interface Hardware	Output Level	Byte Count Includes	Comments
Gigabit Ethernet IQ and IQE PICs	Interface	Queued: 490 bytes per packet, representing 478 bytes of Layer 3 packet + 12 bytes Transmitted: 490 bytes per packet, representing 478 bytes of Layer 3 packet + 12 bytes RED dropped: 496 bytes per packet representing 478 bytes of Layer 3 packet + 18 bytes	The 12 additional bytes include 6 bytes for the destination MAC address + 4 bytes for the VLAN + 2 bytes for the Ethernet type. For RED dropped, 6 bytes are added for the source MAC address.
	Packet forwarding component	Queued: 478 bytes per packet, representing 478 bytes of Layer 3 packet Transmitted: 478 bytes per packet, representing 478 bytes of Layer 3 packet	–

Copyright © 2019, Juniper Networks, Inc.

Table 176: Byte Count by Interface Hardware (continued)

Interface Hardware	Output Level	Byte Count Includes	Comments
Non-IQ PIC	Interface	T Series, TX Series, T1600, and MX Series routers: • Queued: 478 bytes of Layer 3 packet. • Transmitted: 478 bytes of Layer 3 packet. T4000 routers with Type 5 FPCs : • Queued: 478 bytes of Layer 3 packet + the full Layer 2 overhead including 4 bytes CRC + the full Layer 1 overhead 8 bytes preamble + 12 bytes Inter frame Gap. • Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead including 4 bytes CRC + the full Layer 1 overhead 8 bytes preamble + 12 bytes Interframe Gap. M Series routers: • Queued: 478 bytes of Layer 3 packet. • Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead. PTX Series Packet Transport Routers: • Queued: The sum of the transmitted bytes and the RED dropped bytes. • Transmitted: Full Layer 2 overhead (including all L2 encapsulation and CRC) + 12 inter-packet gap + 8 for the preamble. • RED dropped: Full Layer 2 overhead (including all L2 encapsulation and CRC) + 12 inter-packet gap + 8 for the preamble (does not include the VLAN header or MPLS pushed bytes).	The Layer 2 overhead is 14 bytes for non-VLAN traffic and 18 bytes for VLAN traffic.
IQ and IQE PICs with a SONET/SDH interface	Interface	Queued: 482 bytes per packet, representing 478 bytes of Layer 3 packet + 4 bytes Transmitted: 482 bytes per packet, representing 478 bytes of Layer 3 packet + 4 bytes RED dropped: 482 bytes per packet, representing 478 bytes of Layer 3 packet + 4 bytes	The additional 4 bytes are for the Layer 2 Point-to-Point Protocol (PPP) header.
	Packet forwarding component	Queued: 478 bytes per packet, representing 478 bytes of Layer 3 packet Transmitted: 486 bytes per packet, representing 478 bytes of Layer 3 packet + 8 bytes	For transmitted packets, the additional 8 bytes includes 4 bytes for the PPP header and 4 bytes for a cookie.

Copyright © 2019, Juniper Networks, Inc.

Table 176: Byte Count by Interface Hardware (continued)

Interface Hardware	Output Level	Byte Count Includes	Comments
Non-IQ PIC with a SONET/SDH interface	Interface	T Series, TX Series, T1600, and MX Series routers: • Queued: 478 bytes of Layer 3 packet. • Transmitted: 478 bytes of Layer 3 packet. M Series routers: • Queued: 478 bytes of Layer 3 packet. • Transmitted: 483 bytes per packet, representing 478 bytes of Layer 3 packet + 5 bytes • RED dropped: 478 bytes per packet, representing 478 bytes of Layer 3 packet	For transmitted packets, the additional 5 bytes includes 4 bytes for the PPP header and 1 byte for the packet loss priority (PLP).
Interfaces configured with Frame Relay Encapsulation	Interface	The default Frame Relay overhead is 7 bytes. If you configure the Frame Check Sequence (FCS) to 4 bytes, then the overhead increases to 10 bytes.	
1-port 10-Gigabit Ethernet IQ2 and IQ2–E PICs 4-port 1G IQ2 and IQ2-E PICs 8-port 1G IQ2 and IQ2-E PICs	Interface	Queued: 478 bytes of Layer 3 packet + the full Layer 2 overhead including CRC. Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead including CRC.	The Layer 2 overhead is 18 bytes for non-VLAN traffic and 22 bytes for VLAN traffic.
	Packet forwarding component	Queued: 478 bytes of Layer 3 packet. Transmitted: 478 bytes of Layer 3 packet.	–

Sample Output

show interfaces queue (Rate-Limited Interface on a Gigabit Ethernet MIC in an MPC)

The following example shows queue information for the rate-limited interface ge-4/2/0 on a Gigabit Ethernet MIC in an MPC. For rate-limited queues for interfaces hosted on MICs or MPCs, rate-limit packet drops occur prior to packet output queuing. In the command output, the nonzero statistics displayed in the **RL-dropped packets** and **RL-dropped bytes** fields quantify the traffic dropped to rate-limit queue 0 output to 10 percent of 1 gigabyte (100 megabits) per second. Because the RL-dropped traffic is not included in the **Queued** statistics, the statistics displayed for queued traffic are the same as the statistics for transmitted traffic.

```
user@host> show interfaces queue ge-4/2/0
Physical interface: ge-4/2/0, Enabled, Physical link is Up
  Interface index: 203, SNMP ifIndex: 1054
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
```

Copyright © 2019, Juniper Networks, Inc.

```
    Packets              :            131300649              141751 pps
    Bytes                :          11287964840            99793248 bps
  Transmitted:
    Packets              :            131300649              141751 pps
    Bytes                :          11287964840            99793248 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :            205050862              602295 pps
    RL-dropped bytes     :          13595326612            327648832 bps
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
```

show interfaces queue (Aggregated Ethernet on a T320 Router)

The following example shows that the aggregated Ethernet interface, **ae1**, has traffic on queues **af1** and **af12**:

```
user@host> show interfaces queue ae1
```

```
Physical interface: ae1, Enabled, Physical link is Up
Interface index: 158, SNMP ifIndex: 33 Forwarding classes: 8 supported, 8 in use
Output queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: be
  Queued:
    Packets              :                    5                    0 pps
    Bytes                :                  242                    0 bps
  Transmitted:
    Packets              :                    5                    0 pps
    Bytes                :                  242                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 1, Forwarding classes: af1
  Queued:
    Packets              :             42603765               595484 pps
    Bytes                :           5453281920            609776496 bps
  Transmitted:
    Packets              :             42603765               595484 pps
    Bytes                :           5453281920            609776496 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 2, Forwarding classes: ef1
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
  Transmitted:
```


Copyright © 2019, Juniper Networks, Inc.

```
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 3, Forwarding classes: nc
  Queued:
    Packets              :                   45                    0 pps
    Bytes                :                 3930                    0 bps
  Transmitted:
    Packets              :                   45                    0 pps
    Bytes                :                 3930                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 4, Forwarding classes: af11
  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  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 5, Forwarding classes: ef11
  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  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 6, Forwarding classes: af12
  Queued:
    Packets              :             31296413               437436 pps
    Bytes                :           4005940864            447935200 bps
  Transmitted:
    Packets              :             31296413               437436 pps
    Bytes                :           4005940864            447935200 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 7, Forwarding classes: nc2
  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  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
```

show interfaces queue (Gigabit Ethernet on a T640 Router)

```
user@host> show interfaces queue
```

```
Physical interface: ge-7/0/1, Enabled, Physical link is Up
   Interface index: 150, SNMP ifIndex: 42
 Forwarding classes: 8 supported, 8 in use
 Output queues: 8 supported, 8 in use
 Queue: 0, Forwarding classes: be
   Queued:
     Packets              :                   13                     0 pps
     Bytes                :                  622                     0 bps
   Transmitted:
     Packets              :                   13                     0 pps
     Bytes                :                  622                     0 bps
     Tail-dropped packets :                    0                     0 pps
     RED-dropped packets  :                    0                     0 pps
     RED-dropped bytes    :                    0                     0 bps
 Queue: 1, Forwarding classes: af1
   Queued:
     Packets              :            1725947945                372178 pps
     Bytes                :          220921336960              381110432 bps
   Transmitted:
     Packets              :            1725947945                372178 pps
     Bytes                :          220921336960              381110432 bps
     Tail-dropped packets :                    0                     0 pps
     RED-dropped packets  :                    0                     0 pps
     RED-dropped bytes    :                    0                     0 bps
 Queue: 2, Forwarding classes: ef1
   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  :                    0                     0 pps
     RED-dropped bytes    :                    0                     0 bps
 Queue: 3, Forwarding classes: nc
   Queued:
     Packets              :                  571                     0 pps
     Bytes                :                49318                   336 bps
   Transmitted:
     Packets              :                  571                     0 pps
     Bytes                :                49318                   336 bps
     Tail-dropped packets :                    0                     0 pps
     RED-dropped packets  :                    0                     0 pps
     RED-dropped bytes    :                    0                     0 bps
```

show interfaces queue aggregate (Gigabit Ethernet Enhanced DPC)

```
user@host> show interfaces queue ge-2/2/9 aggregate
```

```
Physical interface: ge-2/2/9, Enabled, Physical link is Up
  Interface index: 238, SNMP ifIndex: 71
Forwarding classes: 16 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :             148450735                947295 pps
    Bytes                :            8016344944             409228848 bps
  Transmitted:
    Packets              :              76397439                487512 pps
```


Copyright © 2019, Juniper Networks, Inc.

```
    Bytes                :            4125461868             210602376 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :              72053285                459783 pps
     Low                 :              72053285                459783 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :            3890877444             198626472 bps
     Low                 :            3890877444             198626472 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  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
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              :             410278257                473940 pps
    Bytes                :           22156199518             204742296 bps
  Transmitted:
    Packets              :               4850003                  4033 pps
    Bytes                :             261900162               1742256 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :             405425693                469907 pps
     Low                 :             405425693                469907 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :           21892988124             203000040 bps
     Low                 :           21892988124             203000040 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 3, Forwarding classes: network-control
  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
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
```

Copyright © 2019, Juniper Networks, Inc.

```
    High                :                    0                    0 pps
   RED-dropped bytes    :                    0                    0 bps
    Low                 :                    0                    0 bps
    Medium-low          :                    0                    0 bps
    Medium-high         :                    0                    0 bps
    High                :                    0                    0 bps
Forwarding classes: 16 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets             :             76605230               485376 pps
    Bytes               :           5209211400            264044560 bps
  Transmitted:
    Packets             :             76444631               484336 pps
    Bytes               :           5198235612            263478800 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :               160475                 1040 pps
    Low                 :               160475                 1040 pps
    Medium-low          :                    0                    0 pps
    Medium-high         :                    0                    0 pps
    High                :                    0                    0 pps
    RED-dropped bytes    :             10912300               565760 bps
    Low                 :             10912300               565760 bps
    Medium-low          :                    0                    0 bps
    Medium-high         :                    0                    0 bps
    High                :                    0                    0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  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
    Low                 :                    0                    0 pps
    Medium-low          :                    0                    0 pps
    Medium-high         :                    0                    0 pps
    High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
    Low                 :                    0                    0 bps
    Medium-low          :                    0                    0 bps
    Medium-high         :                    0                    0 bps
    High                :                    0                    0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets             :              4836136                 3912 pps
    Bytes               :            333402032              2139056 bps
  Transmitted:
    Packets             :              3600866                 1459 pps
    Bytes               :            244858888               793696 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :              1225034                 2450 pps
    Low                 :              1225034                 2450 pps
    Medium-low          :                    0                    0 pps
    Medium-high         :                    0                    0 pps
    High                :                    0                    0 pps
    RED-dropped bytes    :             83302312              1333072 bps
    Low                 :             83302312              1333072 bps
    Medium-low          :                    0                    0 bps
```


Copyright © 2019, Juniper Networks, Inc.

```
    Medium-high         :                    0                    0 bps
    High                :                    0                    0 bps
Queue: 3, Forwarding classes: network-control
  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
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps

Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets             :             77059796               486384 pps
    Bytes               :           3544750624            178989576 bps
  Transmitted:
    Packets             :             77059797               486381 pps
    Bytes               :           3544750670            178988248 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  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  :                    0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
Queue: 2, Forwarding classes: assured-forwarding
```

```
  Queued:
    Packets              :              4846580                3934 pps
    Bytes                :            222942680             1447768 bps
  Transmitted:
    Packets              :              4846580                3934 pps
    Bytes                :            222942680             1447768 bps
    Tail-dropped packets :                    0                   0 pps
    RED-dropped packets  :                    0                   0 pps
     Low                 :                    0                   0 pps
     Medium-low          :                    0                   0 pps
     Medium-high         :                    0                   0 pps
     High                :                    0                   0 pps
    RED-dropped bytes    :                    0                   0 bps
     Low                 :                    0                   0 bps
     Medium-low          :                    0                   0 bps
     Medium-high         :                    0                   0 bps
     High                :                    0                   0 bps
Queue: 3, Forwarding classes: network-control
  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  :                    0                   0 pps
     Low                 :                    0                   0 pps
     Medium-low          :                    0                   0 pps
     Medium-high         :                    0                   0 pps
     High                :                    0                   0 pps
    RED-dropped bytes    :                    0                   0 bps
     Low                 :                    0                   0 bps
     Medium-low          :                    0                   0 bps
     Medium-high         :                    0                   0 bps
     High                :                    0                   0 bps
```

show interfaces queue (Gigabit Ethernet IQ2 PIC)

```
user@host> show interfaces queue ge-7/1/3
```

```
Physical interface: ge-7/1/3, Enabled, Physical link is Up
  Interface index: 170, SNMP ifIndex: 70 Forwarding classes: 16 supported, 4 in
use Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :             418390039                  10 pps
    Bytes                :           38910269752                7440 bps
  Transmitted:
    Packets              :             418390039                  10 pps
    Bytes                :           38910269752                7440 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                   0 pps
    RED-dropped bytes    :                    0                   0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
  Transmitted:
    Packets              :                    0                   0 pps
```


Copyright © 2019, Juniper Networks, Inc.

```
    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: assured-forwarding
  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: network-control
  Queued:
    Packets             :                    7055                       1 pps
    Bytes               :                  451552                     512 bps
  Transmitted:
    Packets             :                    7055                       1 pps
    Bytes               :                  451552                     512 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                       0                       0 pps
    RED-dropped bytes    :                       0                       0 bps
Forwarding classes: 16 supported, 4 in use Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets             :                    1031                       0 pps
    Bytes               :                  143292                       0 bps
  Transmitted:
    Packets             :                    1031                       0 pps
    Bytes               :                  143292                       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: expedited-forwarding
  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: assured-forwarding
  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: network-control
  Queued:
    Packets              :                 77009                    11 pps
    Bytes                :               6894286                  7888 bps
  Transmitted:
    Packets              :                 77009                    11 pps
    Bytes                :               6894286                  7888 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, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :                  1031                     0 pps
    Bytes                :                147328                     0 bps
  Transmitted:
    Packets              :                  1031                     0 pps
    Bytes                :                147328                     0 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  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  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets :                     0                     0 pps
```


Copyright © 2019, Juniper Networks, Inc.

```
    RED-dropped packets  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 bps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                 94386                    12 pps
    Bytes                :              13756799                  9568 bps
  Transmitted:
    Packets              :                 94386                    12 pps
    Bytes                :              13756799                  9568 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps
```

show interfaces queue both-ingress-egress (Gigabit Ethernet IQ2 PIC)

```
user@host> show interfaces queue ge-6/2/0 both-ingress-egress
Physical interface: ge-6/2/0, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              : Not Available
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                   254                     0 pps
    Bytes                :                 16274                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              : Not Available
    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: assured-forwarding
  Queued:
```

```
    Packets              : Not Available
    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: network-control
  Queued:
    Packets              : Not Available
    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: 8 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              : Not Available
    Bytes                :                       0                       0 bps
  Transmitted:
    Packets              :                       3                       0 pps
    Bytes                :                     126                       0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                       0                       0 pps
    RED-dropped bytes    :                       0                       0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              : Not Available
    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: assured-forwarding
  Queued:
    Packets              : Not Available
    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: network-control
  Queued:
    Packets              : Not Available
    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
```


Copyright © 2019, Juniper Networks, Inc.

```
Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :              80564692                    0 pps
    Bytes                :            3383717100                    0 bps
  Transmitted:
    Packets              :              80564692                    0 pps
    Bytes                :            3383717100                    0 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :              80564685                    0 pps
    Bytes                :            3383716770                    0 bps
  Transmitted:
    Packets              :              80564685                    0 pps
    Bytes                :            3383716770                    0 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
Queue: 2, Forwarding classes: assured-forwarding
  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  :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                  9397                    0 pps
    Bytes                :               3809052                  232 bps
  Transmitted:
    Packets              :                  9397                    0 pps
    Bytes                :               3809052                  232 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
```

show interfaces queue ingress (Gigabit Ethernet IQ2 PIC)

```
user@host> show interfaces queue ge-6/2/0 ingress
Physical interface: ge-6/2/0, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              : Not Available
    Bytes                :                     0                    0 bps
  Transmitted:
    Packets              :                   288                    0 pps
    Bytes                :                 18450                    0 bps
    Tail-dropped packets : Not Available
```

```
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              : Not Available
    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: assured-forwarding
  Queued:
    Packets              : Not Available
    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: network-control
  Queued:
    Packets              : Not Available
    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
```

show interfaces queue egress (Gigabit Ethernet IQ2 PIC)

```
user@host> show interfaces queue ge-6/2/0 egress
Physical interface: ge-6/2/0, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              : Not Available
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     3                     0 pps
    Bytes                :                   126                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              : Not Available
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
```


Copyright © 2019, Juniper Networks, Inc.

```
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              : Not Available
    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: network-control
  Queued:
    Packets              : Not Available
    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
Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :              80564692                     0 pps
    Bytes                :            3383717100                     0 bps
  Transmitted:
    Packets              :              80564692                     0 pps
    Bytes                :            3383717100                     0 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :              80564685                     0 pps
    Bytes                :            3383716770                     0 bps
  Transmitted:
    Packets              :              80564685                     0 pps
    Bytes                :            3383716770                     0 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  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  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                  9538                     0 pps
    Bytes                :               3819840                     0 bps
  Transmitted:
    Packets              :                  9538                     0 pps
```

```
    Bytes               :              3819840                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
```

show interfaces queue remaining-traffic (Gigabit Ethernet Enhanced DPC)

```
user@host> show interfaces queue ge-2/2/9 remaining-traffic
Physical interface: ge-2/2/9, Enabled, Physical link is Up
  Interface index: 238, SNMP ifIndex: 71
Forwarding classes: 16 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :            110208969               472875 pps
    Bytes                :           5951284434            204282000 bps
  Transmitted:
    Packets              :            110208969               472875 pps
    Bytes                :           5951284434            204282000 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  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
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 2, Forwarding classes: assured-forwarding
  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
     Low                 :                    0                    0 pps
```


Copyright © 2019, Juniper Networks, Inc.

```
    Medium-low          :                      0                      0 pps
    Medium-high         :                      0                      0 pps
    High                :                      0                      0 pps
   RED-dropped bytes    :                      0                      0 bps
    Low                 :                      0                      0 bps
    Medium-low          :                      0                      0 bps
    Medium-high         :                      0                      0 bps
    High                :                      0                      0 bps
Queue: 3, Forwarding classes: network-control
  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
    Low                 :                      0                      0 pps
    Medium-low          :                      0                      0 pps
    Medium-high         :                      0                      0 pps
    High                :                      0                      0 pps
   RED-dropped bytes    :                      0                      0 bps
    Low                 :                      0                      0 bps
    Medium-low          :                      0                      0 bps
    Medium-high         :                      0                      0 bps
    High                :                      0                      0 bps
Forwarding classes: 16 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets             :              109355853                 471736 pps
    Bytes               :             7436199152              256627968 bps
  Transmitted:
    Packets             :              109355852                 471736 pps
    Bytes               :             7436198640              256627968 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                      0                      0 pps
    Low                 :                      0                      0 pps
    Medium-low          :                      0                      0 pps
    Medium-high         :                      0                      0 pps
    High                :                      0                      0 pps
   RED-dropped bytes    :                      0                      0 bps
    Low                 :                      0                      0 bps
    Medium-low          :                      0                      0 bps
    Medium-high         :                      0                      0 bps
    High                :                      0                      0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  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
    Low                 :                      0                      0 pps
    Medium-low          :                      0                      0 pps
    Medium-high         :                      0                      0 pps
    High                :                      0                      0 pps
   RED-dropped bytes    :                      0                      0 bps
```

```
    Low                 :                       0                       0 bps
    Medium-low          :                       0                       0 bps
    Medium-high         :                       0                       0 bps
    High                :                       0                       0 bps
Queue: 2, Forwarding classes: assured-forwarding
  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
     Low                :                       0                       0 pps
     Medium-low         :                       0                       0 pps
     Medium-high        :                       0                       0 pps
     High               :                       0                       0 pps
    RED-dropped bytes    :                       0                       0 bps
     Low                :                       0                       0 bps
     Medium-low         :                       0                       0 bps
     Medium-high        :                       0                       0 bps
     High               :                       0                       0 bps
Queue: 3, Forwarding classes: network-control
  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
     Low                :                       0                       0 pps
     Medium-low         :                       0                       0 pps
     Medium-high        :                       0                       0 pps
     High               :                       0                       0 pps
    RED-dropped bytes    :                       0                       0 bps
     Low                :                       0                       0 bps
     Medium-low         :                       0                       0 bps
     Medium-high        :                       0                       0 bps
     High               :                       0                       0 bps
```

show interfaces queue (Channelized OC12 IQE Type 3 PIC in SONET Mode)

```
user@host> show interfaces queue t3-1/1/0:7
Physical interface: t3-1/1/0:7, Enabled, Physical link is Up

  Interface index: 192, SNMP ifIndex: 1948

  Description: full T3 interface connect to 6ce13 t3-3/1/0:7 for FR testing -
Lam

Forwarding classes: 16 supported, 9 in use

Egress queues: 8 supported, 8 in use

Queue: 0, Forwarding classes: DEFAULT

  Queued:
```


Copyright © 2019, Juniper Networks, Inc.

```
    Packets              :                 214886                13449 pps
    Bytes                :                9884756              5164536 bps
  Transmitted:
    Packets              :                 214886                13449 pps
    Bytes                :                9884756              5164536 bps
    Tail-dropped packets :                      0                    0 pps
    RED-dropped packets  :                      0                    0 pps
     Low                 :                      0                    0 pps
     Medium-low          :                      0                    0 pps
     Medium-high         :                      0                    0 pps
     High                :                      0                    0 pps
    RED-dropped bytes    :                      0                    0 bps
     Low                 :                      0                    0 bps
     Medium-low          :                      0                    0 bps
     Medium-high         :                      0                    0 bps
     High                :                      0                    0 bps
Queue: 1, Forwarding classes: REALTIME
  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  :                      0                    0 pps
     Low                 :                      0                    0 pps
     Medium-low          :                      0                    0 pps
     Medium-high         :                      0                    0 pps
     High                :                      0                    0 pps
    RED-dropped bytes    :                      0                    0 bps
```

```
    Low                 :                    0                    0 bps
    Medium-low          :                    0                    0 bps
    Medium-high         :                    0                    0 bps
    High                :                    0                    0 bps
Queue: 2, Forwarding classes: PRIVATE
  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  :                   0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes    :                   0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
Queue: 3, Forwarding classes: CONTROL
  Queued:
    Packets             :                   60                    0 pps
    Bytes               :                 4560                    0 bps
  Transmitted:
    Packets             :                   60                    0 pps
    Bytes               :                 4560                    0 bps
    Tail-dropped packets :                   0                    0 pps
```


Copyright © 2019, Juniper Networks, Inc.

```
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 4, Forwarding classes: CLASS_B_OUTPUT
  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  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 5, Forwarding classes: CLASS_C_OUTPUT
  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  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 6, Forwarding classes: CLASS_V_OUTPUT
  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  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
```


Copyright © 2019, Juniper Networks, Inc.

```
    Low                :                    0                    0 bps
    Medium-low         :                    0                    0 bps
    Medium-high        :                    0                    0 bps
    High               :                    0                    0 bps
Queue: 7, Forwarding classes: CLASS_S_OUTPUT, GETS
  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  :                  0                    0 pps
     Low               :                    0                    0 pps
     Medium-low        :                    0                    0 pps
     Medium-high       :                    0                    0 pps
     High              :                    0                    0 pps
    RED-dropped bytes  :                    0                    0 bps
     Low               :                    0                    0 bps
     Medium-low        :                    0                    0 bps
     Medium-high       :                    0                    0 bps
     High              :                    0                    0 bps


Packet Forwarding Engine Chassis Queues:
Queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DEFAULT
  Queued:
    Packets            :               371365                23620 pps
    Bytes              :             15597330              7936368 bps
  Transmitted:
```

```
    Packets              :                371365                23620 pps
    Bytes                :              15597330              7936368 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
     Low                 :                     0                    0 pps
     Medium-low          :                     0                    0 pps
     Medium-high         :                     0                    0 pps
     High                :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
     Low                 :                     0                    0 bps
     Medium-low          :                     0                    0 bps
     Medium-high         :                     0                    0 bps
     High                :                     0                    0 bps
Queue: 1, Forwarding classes: REALTIME
  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  :                     0                    0 pps
     Low                 :                     0                    0 pps
     Medium-low          :                     0                    0 pps
     Medium-high         :                     0                    0 pps
     High                :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
     Low                 :                     0                    0 bps
     Medium-low          :                     0                    0 bps
     Medium-high         :                     0                    0 bps
```


Copyright © 2019, Juniper Networks, Inc.

```
    High                :                    0                    0 bps
Queue: 2, Forwarding classes: PRIVATE
  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  :                   0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
Queue: 3, Forwarding classes: CONTROL
  Queued:
    Packets             :                32843                    0 pps
    Bytes               :              2641754                   56 bps
  Transmitted:
    Packets             :                32843                    0 pps
    Bytes               :              2641754                   56 bps
    Tail-dropped packets :                   0                    0 pps
    RED-dropped packets  :                   0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
```

```
    Medium-high         :                    0                    0 pps
    High                :                    0                    0 pps
   RED-dropped bytes    :                    0                    0 bps
    Low                 :                    0                    0 bps
    Medium-low          :                    0                    0 bps
    Medium-high         :                    0                    0 bps
    High                :                    0                    0 bps
Queue: 4, Forwarding classes: CLASS_B_OUTPUT
  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  :                   0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
Queue: 5, Forwarding classes: CLASS_C_OUTPUT
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
```


Copyright © 2019, Juniper Networks, Inc.

```
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 6, Forwarding classes: CLASS_V_OUTPUT
  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  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
```

Copyright © 2019, Juniper Networks, Inc.

```
    High                :                    0                    0 bps
Queue: 7, Forwarding classes: CLASS_S_OUTPUT, GETS
  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  :                   0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 bps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
```

show interfaces queue (QFX Series)

```
user@switch> show interfaces queue xe-0/0/15
  Physical interface: xe-0/0/15, Enabled, Physical link is Up
    Interface index: 49165, SNMP ifIndex: 539
  Forwarding classes: 12 supported, 8 in use
  Egress queues: 12 supported, 8 in use
  Queue: 0, Forwarding classes: best-effort
    Queued:
      Packets              :                    0                    0 pps
      Bytes                :                    0                    0 bps
    Transmitted:
      Packets              :                    0                    0 pps
      Bytes                :                    0                    0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                    0                    0 pps
      Total-dropped bytes  :                    0                    0 bps
  Queue: 3, Forwarding classes: fcoe
```


Copyright © 2019, Juniper Networks, Inc.

```
    Queued:
      Packets              :                     0                     0 pps
      Bytes                :                     0                     0 bps
    Transmitted:
      Packets              :                     0                     0 pps
      Bytes                :                     0                     0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                     0                     0 pps
      Total-dropped bytes  :                     0                     0 bps
0 bps
   Queue: 4, Forwarding classes: no-loss
    Queued:
      Packets              :                     0                     0 pps
      Bytes                :                     0                     0 bps
    Transmitted:
      Packets              :                     0                     0 pps
      Bytes                :                     0                     0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                     0                     0 pps
      Total-dropped bytes  :                     0                     0 bps
   Queue: 7, Forwarding classes: network-control
    Queued:
      Packets              :                     0                     0 pps
      Bytes                :                     0                     0 bps
    Transmitted:
      Packets              :                     0                     0 pps
      Bytes                :                     0                     0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                     0                     0 pps
      Total-dropped bytes  :                     0                     0 bps
   Queue: 8, Forwarding classes: mcast
    Queued:
      Packets              :                     0                     0 pps
      Bytes                :                     0                     0 bps
    Transmitted:
      Packets              :                     0                     0 pps
      Bytes                :                     0                     0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                     0                     0 pps
      Total-dropped bytes  :                     0                     0 bps
```

show interfaces queue l2-statistics (lsq interface)

```
user@switch> show interfaces queue lsq-2/2/0.2 l2-statistics
Logical interface lsq-2/2/0.2 (Index 69) (SNMP ifIndex 1598)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: be
  Queued:
    Packets              :                     1                     0 pps
    Bytes                :                  1001                     0 bps
  Transmitted:
    Packets              :                     5                      0 pps
    Bytes                :                  1062                     0 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
```

```
Queue: 1, Forwarding classes: ef
  Queued:
    Packets              :                    1                    0 pps
    Bytes                :                 1500                    0 bps
  Transmitted:
    Packets              :                    6                     0 pps
    Bytes                :                 1573                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 2, Forwarding classes: af
  Queued:
    Packets              :                    1                    0 pps
    Bytes                :                  512                    0 bps
  Transmitted:
    Packets              :                    3                     0 pps
    Bytes                :                  549                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 3, Forwarding classes: nc
  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  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
=========
```

show interfaces queue lsq (lsq-ifd)

```
user@switch> show interfaces queue lsq-1/0/0
```

```
Logical interface lsq-1/0/0 (Index 348) (SNMP ifIndex 660)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: be
  Queued:
    Packets              :                55576                 1206 pps
    Bytes                :             29622008              5145472 bps
  Transmitted:
    Packets              :                55576                 1206 pps
    Bytes                :             29622008              5145472 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
```


Copyright © 2019, Juniper Networks, Inc.

```
    High                :                    0                    0 bps
Queue: 1, Forwarding classes: ef
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
    Tail-dropped packets :                   0                    0 pps
    RL-dropped packets  :                    0                    0 pps
    RL-dropped bytes    :                    0                    0 bps
    RED-dropped packets :                    0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
Queue: 2, Forwarding classes: af
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
    Tail-dropped packets :                   0                    0 pps
    RL-dropped packets  :                    0                    0 pps
    RL-dropped bytes    :                    0                    0 bps
    RED-dropped packets :                    0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
Queue: 3, Forwarding classes: nc
  Queued:
    Packets             :                22231                  482 pps
    Bytes               :             11849123              2057600 bps
  Transmitted:
    Packets             :                22231                  482 pps
    Bytes               :             11849123              2057600 bps
    Tail-dropped packets :                   0                    0 pps
    RL-dropped packets  :                    0                    0 pps
    RL-dropped bytes    :                    0                    0 bps
    RED-dropped packets :                    0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
```

```
    Medium-high         :                    0                    0 bps
    High                :                    0                    0 bps
```

show interfaces queue (Aggregated Ethernet on a MX series Router)

```
user@host> show interfaces queue ae0 remaining-traffic

Physical interface: ae0    , Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 543
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :                   16                    0 pps
    Bytes                :                 1896                    0 bps
  Transmitted:
    Packets              :                   16                    0 pps
    Bytes                :                 1896                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
  Queue-depth bytes      :
    Average              :                    0
    Current              :                    0
    Peak                 :                    0
    Maximum              :            119013376
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
  Queue-depth bytes      :
    Average              :                    0
    Current              :                    0
```


Copyright © 2019, Juniper Networks, Inc.

```
    Peak                :                    0
    Maximum             :                32768
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
    Tail-dropped packets :                   0                    0 pps
    RL-dropped packets  :                    0                    0 pps
    RL-dropped bytes    :                    0                    0 bps
    RED-dropped packets :                    0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
  Queue-depth bytes     :
    Average             :                    0
    Current             :                    0
    Peak                :                    0
    Maximum             :                32768
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
    Tail-dropped packets :                   0                    0 pps
    RL-dropped packets  :                    0                    0 pps
    RL-dropped bytes    :                    0                    0 bps
    RED-dropped packets :                    0                    0 pps
     Low                :                    0                    0 pps
     Medium-low         :                    0                    0 pps
     Medium-high        :                    0                    0 pps
     High               :                    0                    0 pps
    RED-dropped bytes   :                    0                    0 bps
     Low                :                    0                    0 bps
     Medium-low         :                    0                    0 bps
     Medium-high        :                    0                    0 bps
     High               :                    0                    0 bps
  Queue-depth bytes     :
    Average             :                    0
    Current             :                    0
    Peak                :                    0
    Maximum             :              6258688
```

show interfaces queue ge-0/0/0 (EX2200 Switch)

```
user@switch> show interfaces queue ge-0/0/0
```

```
Physical interface: ge-0/0/0, Enabled, Physical link is Down
  Interface index: 130, SNMP ifIndex: 501
Forwarding classes: 16 supported, 4 in use
```

```
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
Queue: 1, Forwarding classes: assured-forwarding
  Queued:
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
Queue: 5, Forwarding classes: expedited-forwarding
  Queued:
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
Queue: 7, Forwarding classes: network-control
  Queued:
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
```

show interfaces queue xe-6/0/39 (Line Card with Oversubscribed Ports in an EX8200 Switch)

```
user@switch> show interfaces queue xe-6/0/39
```

```
Physical interface: xe-6/0/39, Enabled, Physical link is Up
  Interface index: 291, SNMP ifIndex: 1641
Forwarding classes: 16 supported, 7 in use
Ingress queues: 1 supported, 1 in use
  Transmitted:
    Packets              :          337069086018
    Bytes                :        43144843010304
    Tail-dropped packets :            8003867575
PFE chassis queues: 1 supported, 1 in use
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
Forwarding classes: 16 supported, 7 in use
Egress queues: 8 supported, 7 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
  Transmitted:
    Packets              :          334481399932
    Bytes                :        44151544791024
    Tail-dropped packets :                    0
  Queue: 1, Forwarding classes: assured-forwarding
  Queued:
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
  Queue: 2, Forwarding classes: mcast-be
  Queued:
```


Copyright © 2019, Juniper Networks, Inc.

```
  Transmitted:
    Packets              :             274948977
    Bytes                :           36293264964
    Tail-dropped packets :                     0
Queue: 4, Forwarding classes: mcast-ef
  Queued:
  Transmitted:
    Packets              :                     0
    Bytes                :                     0
    Tail-dropped packets :                     0
 Queue: 5, Forwarding classes: expedited-forwarding
  Queued:
  Transmitted:
    Packets              :                     0
    Bytes                :                     0
    Tail-dropped packets :                     0
  Queue: 6, Forwarding classes: mcast-af
  Queued:
  Transmitted:
    Packets              :                     0
    Bytes                :                     0
    Tail-dropped packets :                     0
  Queue: 7, Forwarding classes: network-control
  Queued:
  Transmitted:
    Packets              :                 46714
    Bytes                :               6901326
    Tail-dropped packets :                     0

Packet Forwarding Engine Chassis Queues:
Queues: 8 supported, 7 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
  Transmitted:
    Packets              :          739338141426
    Bytes                :        94635282101928
    Tail-dropped packets :                     0
    RED-dropped packets  :            5606426444
     Low                 :            5606426444
     High                :                     0
    RED-dropped bytes    :          683262846464
     Low                 :          683262846464
     High                :                     0
Queue: 1, Forwarding classes: assured-forwarding
  Queued:
  Transmitted:
    Packets              :                     0
    Bytes                :                     0
    Tail-dropped packets :                     0
    RED-dropped packets  :                     0
     Low                 :                     0
     High                :                     0
    RED-dropped bytes    :                     0
     Low                 :                     0
     High                :                     0
Queue: 2, Forwarding classes: mcast-be
  Queued:
  Transmitted:
    Packets              :                     0
    Bytes                :                     0
```

```
    Tail-dropped packets :                      0
    RED-dropped packets  :                      0
     Low                 :                      0
     High                :                      0
    RED-dropped bytes    :                      0
     Low                 :                      0
     High                :                      0
Queue: 4, Forwarding classes: mcast-ef
  Queued:
  Transmitted:
    Packets              :                      0
    Bytes                :                      0
    Tail-dropped packets :                      0
    RED-dropped packets  :                      0
     Low                 :                      0
     High                :                      0
    RED-dropped bytes    :                      0
     Low                 :                      0
     High                :                      0
Queue: 5, Forwarding classes: expedited-forwarding
  Queued:
  Transmitted:
    Packets              :                      0
    Bytes                :                      0
    Tail-dropped packets :                      0
    RED-dropped packets  :                      0
     Low                 :                      0
     High                :                      0
    RED-dropped bytes    :                      0
     Low                 :                      0
     High                :                      0
Queue: 6, Forwarding classes: mcast-af
  Queued:
  Transmitted:
    Packets              :                      0
    Bytes                :                      0
    Tail-dropped packets :                      0
    RED-dropped packets  :                      0
     Low                 :                      0
     High                :                      0
    RED-dropped bytes    :                      0
     Low                 :                      0
     High                :                      0
Queue: 7, Forwarding classes: network-control
  Queued:
  Transmitted:
    Packets              :                  97990
    Bytes                :               14987506
    Tail-dropped packets :                      0
    RED-dropped packets  :                      0
     Low                 :                      0
     High                :                      0
    RED-dropped bytes    :                      0
     Low                 :                      0
     High                :                      0
```


Copyright © 2019, Juniper Networks, Inc.

show interfaces queue xe-0/0/2 buffer-occupancy (QFX5000 Series switch)

```
user@switch> show interfaces queue xe-0/0/2 buffer-occupancy

Physical interface: xe-0/0/2, Enabled, Physical link is Up
  Interface index: 689, SNMP ifIndex: 514
Forwarding classes: 12 supported, 5 in use
Egress queues: 12 supported, 5 in use
            Queue: 0, Forwarding classes: fc0
            Queue-depth bytes  :
            Peak               : 1786720
            Queue: 3, Forwarding classes: fcoe
            Queue-depth bytes  :
            Peak               : 0
            Queue: 4, Forwarding classes: no-loss
            Queue-depth bytes  :
            Peak               : 0
            Queue: 7, Forwarding classes: network-control
            Queue-depth bytes  :
            Peak               : 416
            Queue: 8, Forwarding classes: mcast
            Queue-depth bytes  :
            Peak               : 0
```

Copyright © 2019, Juniper Networks, Inc.

show interfaces voq

Syntax

```
show interfaces voq interface-name
<forwarding-class forwarding-class-name>
<non-zero>
<source-fpc source-fpc-number>
```

Syntax (Junos OS Evolved)

```
show interfaces voq interface-name
<forwarding-class forwarding-class-name>
<non-zero>
```

Release Information

Command introduced in Junos OS Release 14.1 for the PTX Series Routers
Command introduced in Junos OS Release 15.1X53-D20 for QFX10000 switches.
Option **source-fpc** deprecated in Junos OS Evolved Release 19.1R1.

Description

Display the random early detection (RED) drop statistics from all ingress Packet Forwarding Engines associated with the specified physical egress interface. In the VOQ architecture, egress output queues (shallow buffers) buffer data in virtual queues on ingress Packet Forwarding Engines. In cases of congestion, you can use this command to identify which ingress Packet Forwarding Engine is the source of RED-dropped packets contributing to congestion.

NOTE: On the PTX Series routers and QFX10000 switches, these statistics include tail-dropped packets.

On platforms released in Junos OS Evolved Release 19.1R1 (PTX10003), there are only logical FPCs. Therefore, the **source-fpc** option is not required.

Options

interface *interface-name*—Display the ingress VOQ RED drop statistics for the specified egress interface.

forwarding-class *forwarding-class-name*—Display VOQ RED drop statistics for a specified forwarding class.

non-zero—Display only non-zero VOQ RED drop statistics counters.

source-fpc *source-fpc-number*—Display VOQ RED drop statistics for the specified source FPC.

Additional Information

- On PTX Series routers, you can display VOQ statistics for only the WAN physical interface.
- VOQ statistics for aggregated physical interfaces are not supported. Statistics for an aggregated interface are the summation of the queue statistics of the child links of that aggregated interface. You can use the **show interfaces queue** command to identify

Copyright © 2019, Juniper Networks, Inc.

the child link which is experiencing congestion and then view the VOQ statistics on the respective child link using the **show interfaces voq** command.

For information on virtual output queuing on PTX routers, see "Understanding Virtual Output Queues on PTX Series Packet Transport Routers" on page 607. For information on virtual output queueing on QFX10000 switches, see *Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches*.

Required Privilege Level view

Related Documentation

- Understanding Virtual Output Queues on PTX Series Packet Transport Routers on page 607
- *Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches*

List of Sample Output

show interfaces voq (For a Specific Physical Interface) (PTX Series Routers) on page 1470
show interfaces voq (For a Specific Physical Interface) (QFX10000 Switches) on page 1475
show interfaces voq et-7/0/0 (For a Specific Forwarding Class) on page 1477
show interfaces voq et-5/0/12 (For a Specific Source FPC) on page 1478
show interfaces voq et-5/0/12 (For a Specific Forwarding Class and Source FPC) on page 1479
show interfaces voq et-7/0/0 (Non-Zero) on page 1480
show interfaces voq et-7/0/0 (For a Specific Forwarding Class and Non-Zero) on page 1480

Output Fields Table 177 on page 1469 lists the output fields for the show interfaces queue command. Output fields are listed in the approximate order in which they appear.

Table 177: show interfaces voq Output Fields

Field Name	Field Description
Physical interface	Name of the physical interface.
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.
Interface index	Physical interface's index number, which reflects its initialization sequence.
SNMP ifIndex	SNMP index number for the interface.
Queue	Egress queue number.
Forwarding classes	Forwarding class name.
FPC number	Number of the Flexible PIC Concentrator (FPC) located on ingress.

Copyright © 2019, Juniper Networks, Inc.

Table 177: show interfaces voq Output Fields (continued)

Field Name	Field Description
PFE	Number of the Packet Forwarding Engine providing virtual output queues on the ingress.
RED-dropped packets	Number of packets per second (pps) dropped because of random early detection (RED). NOTE: On the PTX Series routers, these statistics include tail-dropped packets.
RED-dropped bytes	Number of bytes per second dropped because of RED. The byte counts vary by interface hardware. NOTE: On the PTX Series routers, these statistics include tail-dropped packets.

Sample Output

show interfaces voq (For a Specific Physical Interface) (PTX Series Routers)

The following example shows ingress RED-dropped statistics for the egress Ethernet interface configured on port 0 of Physical Interface Card (PIC) 0, located on the FPC in slot 7.

The sample output below shows that the cause of the congestion is ingress Packet Forwarding Engine PFE 0, which resides on FPC number 4, as denoted by the count of RED-dropped packets and RED-dropped bytes for egress queue 0, forwarding classes best-effort and egress queue 3, forwarding class network control.

```
user@host> show interfaces voq et-7/0/0
Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 155, SNMP ifIndex: 699

Queue: 0, Forwarding classes: best-effort

FPC number: 1
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 4
    PFE: 0
      RED-dropped packets  :             19969426              2323178 pps
      RED-dropped bytes    :           2196636860           2044397464 bps
```


Copyright © 2019, Juniper Networks, Inc.

```
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              19969424               2321205 pps
      RED-dropped bytes    :            2196636640            2042660808 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 4
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 5
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 6
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 7
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

  FPC number: 7
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

Queue: 1, Forwarding classes: expedited-forwarding

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
```

```
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 3
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps

FPC number: 4
  PFE: 0
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 1
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 2
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 3
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps

FPC number: 6
  PFE: 0
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 1
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 2
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 3
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 4
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 5
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 6
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 7
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps

FPC number: 7
  PFE: 0
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 1
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 2
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 3
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
```


Copyright © 2019, Juniper Networks, Inc.

```
Queue: 2, Forwarding classes: assured-forwarding

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 4
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 4
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 5
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 6
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 7
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 7
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
```

```
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

Queue: 3, Forwarding classes: network-control

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              16338670               1900314 pps
      RED-dropped bytes    :            1797253700            1672276976 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              16338698               1899163 pps
      RED-dropped bytes    :            1797256780            1671263512 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 4
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 5
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 6
```


Copyright © 2019, Juniper Networks, Inc.

```
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 7
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps

FPC number: 7
  PFE: 0
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 1
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 2
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 3
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
```

show interfaces voq (For a Specific Physical Interface) (QFX10000 Switches)

The sample output below shows congestion on ingress PFE 1 on FPC number 0, and on ingress PFE 2 on FPC number 1, as denoted by the count of RED-dropped packets and RED-dropped bytes for best-effort egress queue 0.

```
user@host> show interfaces voq et-1/0/0
```

```
Physical interface: et-1/0/0, Enabled, Physical link is Up
  Interface index: 659, SNMP ifIndex: 539

Queue: 0, Forwarding classes: best-effort

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :             411063248             16891870 pps
      RED-dropped bytes    :           52616095744          17297275600 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :             411063012             16891870 pps
      RED-dropped bytes    :           52616065536          17297275376 bps

Queue: 3, Forwarding classes: fcoe

  FPC number: 0
    PFE: 0
```

```
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

Queue: 4, Forwarding classes: no-loss

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

Queue: 7, Forwarding classes: network-control

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
```


Copyright © 2019, Juniper Networks, Inc.

```
    RED-dropped bytes    :                    0                    0 bps
  PFE: 1
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 2
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
```

show interfaces voq et-7/0/0 (For a Specific Forwarding Class)

```
user@host> show interfaces voq et-7/0/0 forwarding-class best-effort
```

```
Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 155, SNMP ifIndex: 699

Queue: 0, Forwarding classes: best-effort

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 4
    PFE: 0
      RED-dropped packets  :             66604786              2321519 pps
      RED-dropped bytes    :           7326526460           2042936776 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :             66604794               371200 pps
      RED-dropped bytes    :           7326527340            326656000 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 4
      RED-dropped packets  :                    0                    0 pps
```

Copyright © 2019, Juniper Networks, Inc.

```
    RED-dropped bytes    :                    0                    0 bps
  PFE: 5
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 6
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 7
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps

FPC number: 7
  PFE: 0
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 1
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 2
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 3
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
```

show interfaces voq et-5/0/12 (For a Specific Source FPC)

```
user@host> show interfaces voq et-5/0/12 source-fpc 0
```

```
Physical interface: et-5/0/12, Enabled, Physical link is Up
    Interface index: 166, SNMP ifIndex: 1104

  Queue: 0, Forwarding classes: best-effort

    FPC number: 0
      PFE: 0
        RED-dropped packets  :                      0                      0 pps
        RED-dropped bytes    :                      0                      0 bps
      PFE: 1
        RED-dropped packets  :                      0                      0 pps
        RED-dropped bytes    :                      0                      0 bps
      PFE: 2
        RED-dropped packets  :                      0                      0 pps
        RED-dropped bytes    :                      0                      0 bps
      PFE: 3
        RED-dropped packets  :                      0                      0 pps
        RED-dropped bytes    :                      0                      0 bps

  Queue: 1, Forwarding classes: expedited-forwarding

    FPC number: 0
      PFE: 0
        RED-dropped packets  :                      0                      0 pps
        RED-dropped bytes    :                      0                      0 bps
      PFE: 1
        RED-dropped packets  :                      0                      0 pps
        RED-dropped bytes    :                      0                      0 bps
      PFE: 2
        RED-dropped packets  :                      0                      0 pps
        RED-dropped bytes    :                      0                      0 bps
```


Copyright © 2019, Juniper Networks, Inc.

```
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  Queue: 2, Forwarding classes: assured-forwarding

    FPC number: 0
      PFE: 0
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 1
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 2
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 3
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps

  Queue: 3, Forwarding classes: network-control

    FPC number: 0
      PFE: 0
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 1
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 2
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 3
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
```

show interfaces voq et-5/0/12 (For a Specific Forwarding Class and Source FPC)

```
user@host> show interfaces voq et-5/0/12 forwarding-class best-effort source-fpc 5
```

```
Physical interface: et-5/0/12, Enabled, Physical link is Up
    Interface index: 166, SNMP ifIndex: 1104

  Queue: 0, Forwarding classes: best-effort

    FPC number: 5
      PFE: 0
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 1
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 2
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 3
        RED-dropped packets  :                  0                    0 pps
        RED-dropped bytes    :                  0                    0 bps
      PFE: 4
        RED-dropped packets  :                  0                    0 pps
```

```
      RED-dropped bytes    :                    0                    0 bps
    PFE: 5
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 6
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 7
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
```

show interfaces voq et-7/0/0 (Non-Zero)

```
user@host> show interfaces voq et-7/0/0 non-zero

Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 155, SNMP ifIndex: 699

Queue: 0, Forwarding classes: best-effort

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              95862238              2301586 pps
      RED-dropped bytes    :           10544846180           2025396264 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              95866639              2322569 pps
      RED-dropped bytes    :           10545330290           2043860728 bps

Queue: 3, Forwarding classes: network-control

  FPC number: 4
    PFE: 0
      RED-dropped packets  :              78433066              1899727 pps
      RED-dropped bytes    :            8627637260           1671760384 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :              78436704              1900628 pps
      RED-dropped bytes    :            8628037440           1672553432 bps
```

show interfaces voq et-7/0/0 (For a Specific Forwarding Class and Non-Zero)

```
user@host show interfaces voq et-7/0/0 forwarding-class best-effort non-zero
Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 155, SNMP ifIndex: 699

Queue: 0, Forwarding classes: best-effort

  FPC number: 4
    PFE: 0
      RED-dropped packets  :             119540012              2322319 pps
      RED-dropped bytes    :           13149401320           2043640784 bps
```


Copyright © 2019, Juniper Networks, Inc.

```
FPC number: 6
  PFE: 0
    RED-dropped packets  :             119540049              2322988 pps
    RED-dropped bytes    :           13149405390           2044229744 bps
```

Copyright © 2019, Juniper Networks, Inc.

Copyright © 2019, Juniper Networks, Inc.