

# Class of Service Feature Guide for EX9200 Switches

Release

15.1



---

Modified: 2016-04-20

Juniper Networks, Inc.  
1133 Innovation Way  
Sunnyvale, California 94089  
USA  
408-745-2000  
[www.juniper.net](http://www.juniper.net)

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, 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.

*Class of Service Feature Guide for EX9200 Switches*

15.1

Copyright © 2016, 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 <http://www.juniper.net/support/eula.html>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

# Table of Contents

	About the Documentation . . . . .	xxi
	Documentation and Release Notes . . . . .	xxi
	Supported Platforms . . . . .	xxi
	Using the Examples in This Manual . . . . .	xxi
	Merging a Full Example . . . . .	xxii
	Merging a Snippet . . . . .	xxii
	Documentation Conventions . . . . .	xxiii
	Documentation Feedback . . . . .	xxv
	Requesting Technical Support . . . . .	xxv
	Self-Help Online Tools and Resources . . . . .	xxv
	Opening a Case with JTAC . . . . .	xxvi
<b>Part 1</b>	<b>Overview</b>	
<b>Chapter 1</b>	<b>Understanding How Class of Service Alleviates Congestion and Defines Traffic Forwarding Behavior . . . . .</b>	<b>3</b>
	Understanding How Class of Service Manages Congestion and Controls Service	
	Levels in the Network . . . . .	3
	CoS Applications . . . . .	4
	CoS Standards . . . . .	5
	The Junos OS CoS Components Used to Manage Congestion and Control Service	
	Levels . . . . .	5
	Mapping CoS Component Inputs to Outputs . . . . .	9
	Default Junos OS CoS Settings . . . . .	12
<b>Chapter 2</b>	<b>Understanding Class of Service Packet Flow Across a Network . . . . .</b>	<b>15</b>
	How CoS Applies to Packet Flow Across a Network . . . . .	15
	Packet Flow Through the Junos OS CoS Process Overview . . . . .	16
	Packet Flow Within Routers Overview . . . . .	18
<b>Part 2</b>	<b>Configuring Class of Service</b>	
<b>Chapter 3</b>	<b>Assigning Service Levels to Packets Using Behavior Aggregate Classifiers . . . . .</b>	<b>21</b>
	Default Aliases for CoS Value Bit Patterns Overview . . . . .	22
	Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic . . . . .	26
	Default Behavior Aggregate Classification Overview . . . . .	29
	Defining Aliases for CoS Value Bit Patterns . . . . .	30
	Applying Behavior Aggregate Classifiers to Logical Interfaces . . . . .	32
	Using BA Classifiers to Set PLP . . . . .	35
	Configuring Behavior Aggregate Classifiers . . . . .	36
	Default IP Precedence Classifier . . . . .	38

	Default IP Precedence Classifier (ipprec-default) . . . . .	40
	Understanding DSCP Classification for VPLS . . . . .	41
	VPLS and Default CoS Classification . . . . .	41
	Default DSCP and DSCP IPv6 Classifiers . . . . .	42
	Applying DSCP Classifiers to MPLS Traffic . . . . .	43
	Applying a DSCP Classifier to MPLS Packets on a Core-facing Interface . . . . .	44
	Applying a DSCP Classifier to MPLS Traffic for L3VPN/VPLS . . . . .	45
	Default MPLS EXP Classifier . . . . .	47
	Configuring CoS for MPLS Traffic . . . . .	48
	Default MPLS EXP Classifier . . . . .	49
	Applying MPLS EXP Classifiers to Routing Instances . . . . .	50
	Configuring and Applying Custom MPLS EXP Classifiers to Routing Instances . . . . .	51
	Applying Global Classifiers and Wildcard Routing Instances . . . . .	52
	Example: Applying Global MPLS EXP Classifiers to Routing Instances . . . . .	52
	Applying Classifiers by Using Wildcard Routing Instances . . . . .	53
	Verifying the Classifiers Associated with Routing Instances . . . . .	54
	Applying MPLS EXP Classifiers for Explicit-Null Labels . . . . .	55
	Default IEEE 802.1p Classifier . . . . .	56
	DSCP IPv6 Rewrites and Forwarding Class Maps . . . . .	57
	Applying DSCP IPv6 Classifiers . . . . .	58
	Applying MPLS EXP Classifiers to Routing Instances . . . . .	58
	Configuring and Applying Custom MPLS EXP Classifiers to Routing Instances . . . . .	59
	Applying Global Classifiers and Wildcard Routing Instances . . . . .	60
	Example: Applying Global MPLS EXP Classifiers to Routing Instances . . . . .	60
	Applying Classifiers by Using Wildcard Routing Instances . . . . .	62
	Verifying the Classifiers Associated with Routing Instances . . . . .	63
	Applying MPLS EXP Classifiers for Explicit-Null Labels . . . . .	63
	Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets . . . . .	64
	Default IEEE 802.1ad Classifier . . . . .	66
	Configuring and Applying IEEE 802.1ad Classifiers . . . . .	67
	Defining Custom IEEE 802.1ad Maps . . . . .	68
	Applying Custom IEEE 802.1ad Maps . . . . .	68
	Verifying Custom IEEE 802.1ad Map Configuration . . . . .	68
<b>Chapter 4</b>	<b>Assigning Service Levels to Packets Using Multifield Classifiers . . . . .</b>	<b>69</b>
	Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields . . . . .	69
	Configuring Multifield Classifiers . . . . .	70
	Example: Classifying Packets Based on Their Destination Address . . . . .	73
	Example: Configuring and Verifying a Complex Multifield Filter . . . . .	74
	Configuring a Complex Multifield Filter . . . . .	74
	Verifying a Complex Multifield Filter . . . . .	76
<b>Chapter 5</b>	<b>Controlling Access to the Network Using Traffic Policing . . . . .</b>	<b>79</b>
	Controlling Network Access Using Traffic Policing Overview . . . . .	79
	Congestion Management for IP Traffic Flows . . . . .	79
	Traffic Limits . . . . .	80

Traffic Color Marking . . . . .	81
Forwarding Classes and PLP Levels . . . . .	83
Policer Application to Traffic . . . . .	84
Effect of Two-Color Policers on Shaping Rate Changes . . . . .	85
Configuring Two-Color Policers and Shaping Rate Changes . . . . .	86
Configuring Policers Based on Logical Interface Bandwidth . . . . .	87
Example: Configuring a Logical Bandwidth Policer . . . . .	89
Overview of Tricolor Marking Architecture . . . . .	90
Tricolor Marking Limitations . . . . .	91
Configuring Tricolor Marking . . . . .	92
Configuring Single-Rate Tricolor Marking . . . . .	93
Configuring Color-Blind Mode for Single-Rate Tricolor Marking . . . . .	93
Configuring Color-Aware Mode for Single-Rate Tricolor Marking . . . . .	94
Effect on Low PLP of Single-Rate Policer . . . . .	94
Effect on Medium-Low PLP of Single-Rate Policer . . . . .	95
Effect on Medium-High PLP of Single-Rate Policer . . . . .	95
Effect on High PLP of Single-Rate Policer . . . . .	96
Configuring Two-Rate Tricolor Marking . . . . .	96
Configuring Color-Blind Mode for Two-Rate Tricolor Marking . . . . .	96
Configuring Color-Aware Mode for Two-Rate Tricolor Marking . . . . .	97
Effect on Low PLP of Two-Rate Policer . . . . .	97
Effect on Medium-Low PLP of Two-Rate Policer . . . . .	98
Effect on Medium-High PLP of Two-Rate Policer . . . . .	98
Effect on High PLP of Two-Rate Policer . . . . .	99
Enabling Tricolor Marking and Limitations of Three-Color Policers . . . . .	100
Configuring and Applying Tricolor Marking Policers . . . . .	102
Defining a Tricolor Marking Policer . . . . .	102
Applying Tricolor Marking Policers to Firewall Filters . . . . .	104
Applying Firewall Filter Tricolor Marking Policers to Interfaces . . . . .	105
Example: Configuring and Applying a Single-Rate Tricolor Marking Policer . . . . .	106
Applying Tricolor Marking Policers to Firewall Filters . . . . .	107
Example: Applying a Two-Rate Tricolor Marking Policer to a Firewall Filter . . . . .	108
Applying Firewall Filter Tricolor Marking Policers to Interfaces . . . . .	108
Example: Applying a Single-Rate Tricolor Marking Policer to an Interface . . . . .	108
<b>Chapter 6 Defining Forwarding Behavior Based on Forwarding Classes . . . . .</b>	<b>111</b>
Forwarding Classes Overview . . . . .	111
Output Queue Assignments Based on Forwarding Class . . . . .	112
Devices That Support Up to Four Forwarding Classes . . . . .	112
Devices That Support Up to 16 Forwarding Classes . . . . .	113
Default and Configurable Packet Loss Priority Values . . . . .	113

	Configuration Statements Used to Configure and Apply Forwarding	
	Classes . . . . .	113
	Default Forwarding Classes . . . . .	114
	Configuring Forwarding Classes . . . . .	116
	Configuring Up to 16 Forwarding Classes . . . . .	117
	Enabling Eight Queues on Interfaces . . . . .	120
	Multiple Forwarding Classes and Default Forwarding Classes . . . . .	121
	PICs Restricted to Four Queues . . . . .	121
	Examples: Configuring Up to 16 Forwarding Classes . . . . .	122
	Classifying Packets by Egress Interface . . . . .	124
	Forwarding Policy Options Overview . . . . .	126
	Configuring CoS-Based Forwarding . . . . .	126
	Example: Configuring CoS-Based Forwarding . . . . .	129
	Example: Configuring CoS-Based Forwarding for Different Traffic Types . . . . .	132
	Example: Configuring CoS-Based Forwarding for IPv6 . . . . .	132
	Overriding the Input Classification . . . . .	133
	Applying Forwarding Classes to Interfaces . . . . .	134
	Default Routing Engine Protocol Queue Assignments . . . . .	135
	Changing the Default Queuing and Marking of Host Outbound Traffic . . . . .	137
	Assigning Forwarding Class and DSCP Value for Routing Engine-Generated	
	Traffic . . . . .	138
	Managing Congestion by Setting Packet Loss Priority for Different Traffic	
	Flows . . . . .	139
	Example: Overriding the Default PLP on M320 Routers . . . . .	140
	Mapping PLP to RED Drop Profiles . . . . .	140
<b>Chapter 7</b>	<b>Defining Output Queue Properties Using Schedulers . . . . .</b>	<b>143</b>
	Schedulers Overview . . . . .	143
	Default Schedulers Overview . . . . .	145
	Configuring Schedulers . . . . .	146
	Managing Congestion on the Egress Interface by Configuring the Scheduler	
	Buffer Size . . . . .	146
	Configuring Large Delay Buffers for Slower Interfaces . . . . .	148
	Maximum Delay Buffer for NxDSO Interfaces . . . . .	151
	Example: Configuring Large Delay Buffers for Slower Interfaces . . . . .	153
	Enabling and Disabling the Memory Allocation Dynamic per Queue . . . . .	156
	Configuring Scheduler Maps . . . . .	157
	Applying Scheduler Maps to Physical Interfaces . . . . .	158
	Priority Scheduling Overview . . . . .	158
	Applying Scheduler Maps Overview . . . . .	159
	Applying a Shaping Rate to Physical Interfaces Overview . . . . .	160
	Forwarding Classes and Fabric Priority Queues . . . . .	161
	Default Fabric Priority Queuing . . . . .	161
	Overriding Default Fabric Priority Queuing . . . . .	161

<b>Chapter 8</b>	<b>Controlling Bandwidth Using Scheduler Rates</b>	<b>163</b>
	Oversubscribing Interface Bandwidth	163
	Verifying Configuration of Bandwidth Oversubscription	169
	Examples: Oversubscribing Interface Bandwidth	169
	Configuring Scheduler Transmission Rate	172
	Example: Configuring Scheduler Transmission Rate	173
	Allocation of Leftover Bandwidth	174
	Providing a Guaranteed Minimum Rate	175
	Verifying Configuration of Guaranteed Minimum Rate	177
	Example: Providing a Guaranteed Minimum Rate	178
	Bandwidth Sharing on Nonqueuing Packet Forwarding Engines Overview	179
	Configuring Rate Limits on Nonqueuing Packet Forwarding Engines	179
	Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs	181
	Applying Scheduler Maps to Packet Forwarding Component Queues	188
	Applying Custom Schedulers to Packet Forwarding Component Queues	189
	Examples: Scheduling Packet Forwarding Component Queues	190
<b>Chapter 9</b>	<b>Setting Transmission Order Using Scheduler Priorities</b>	<b>195</b>
	Configuring Schedulers for Priority Scheduling	195
	Example: Configuring Priority Scheduling	195
	Strict-High Priority Configuration Overview	196
	Associating Schedulers with Fabric Priorities	197
	Example: Associating a Scheduler with a Fabric Priority	197
<b>Chapter 10</b>	<b>Controlling Congestion Using Scheduler RED Drop Profiles and Buffers</b>	<b>199</b>
	Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers	199
	RED Drop Profiles Overview	200
	Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows	203
	Example: Overriding the Default PLP on M320 Routers	203
	Mapping PLP to RED Drop Profiles	204
	Defining Packet Drop Behavior by Configuring RED Drop Profiles	205
	Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy	206
	Configuring Rewrite Rules Based on PLP	208
	Example: Configuring Weighted RED Buffer Occupancy	208
<b>Chapter 11</b>	<b>Altering Outgoing Packet Headers Using Rewrite Rules to Ensure Forwarding Behavior</b>	<b>211</b>
	Rewriting Packet Header Information Overview	212
	Applying Default Rewrite Rules	213
	Configuring Rewrite Rules	215
	Applying IEEE 802.1p Rewrite Rules to Dual VLAN Tags	216
	Example: Applying an IEEE 802.1p Rewrite Rule to Dual VLAN Tags	217
	Setting IPv6 DSCP and MPLS EXP Values Independently	217
	Classifiers and Rewrite Rules at the Global and Physical Interface Levels Overview	218

	Applying Rewrite Rules to Output Logical Interfaces . . . . .	218
	Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel . . . . .	220
	Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels . . . . .	222
	Rewriting MPLS and IPv4 Packet Headers . . . . .	224
	Example: Rewriting MPLS and IPv4 Packet Headers . . . . .	225
	Example: Simultaneous DSCP and EXP Rewrite . . . . .	227
	Rewriting the EXP Bits of All Three Labels of an Outgoing Packet . . . . .	228
	Example: Rewriting the EXP Bits of All Three Labels of an Outgoing Packet . . . . .	229
	Rewriting IEEE 802.1p Packet Headers with an MPLS EXP Value . . . . .	229
	Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic . . . . .	231
	Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic . . . . .	232
	Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface . . . . .	232
	Setting Ingress DSCP Bits for Multicast Traffic over Layer 3 VPNs . . . . .	233
	Example: Per-Node Rewriting of EXP Bits . . . . .	234
<b>Part 3</b>	<b>Configuring Platform-Specific Functionality</b>	
<b>Chapter 12</b>	<b>Configuring Class of Service on EX Series Ethernet Switches . . . . .</b>	<b>239</b>
	Classifiers and Rewrite Rules at the Global and Physical Interface Levels Overview . . . . .	239
	Configuring the Shaping Rate for Physical Interfaces . . . . .	240
	Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels . . . . .	241
<b>Part 4</b>	<b>Configuring Line Card-Specific and Interface-Specific Functionality</b>	
<b>Chapter 13</b>	<b>Feature Support for Line Cards and Interfaces . . . . .</b>	<b>245</b>
	Interface Types That Do Not Support CoS . . . . .	245
<b>Chapter 14</b>	<b>Configuring Class of Service for Tunnels . . . . .</b>	<b>247</b>
	CoS for Tunnels Overview . . . . .	247
	Configuring CoS for Tunnels . . . . .	248
	Tunneling and BA Classifiers . . . . .	248
	Example: Configuring CoS for Tunnels . . . . .	249
	Example: Configuring a GRE Tunnel to Copy ToS Bits to the Outer IP Header . . . . .	252
<b>Chapter 15</b>	<b>Configuring Class of Service on IQ and Enhanced IQ (IQE) PICs . . . . .</b>	<b>253</b>
	Simple Filters Overview . . . . .	253
	Example: Configuring a Simple Filter . . . . .	254
	BA Classifiers and ToS Translation Tables . . . . .	255
	CoS for L2TP Tunnels on Ethernet Interface Overview . . . . .	255
	Configuring CoS for L2TP Tunnels on Ethernet Interfaces . . . . .	256
	Configuring LNS CoS for Link Redundancy . . . . .	257
	Example: Configuring L2TP LNS CoS Support for Link Redundancy . . . . .	258

<b>Chapter 16</b>	<b>Configuring Class of Service on Aggregated, Channelized, and Gigabit Ethernet Interfaces . . . . .</b>	<b>263</b>
	Limitations on CoS for Aggregated Interfaces . . . . .	263
	Configuring Per-Unit Schedulers for Channelized Interfaces . . . . .	265
	Configuring Schedulers on Aggregated Interfaces . . . . .	268
	Policer Support for Aggregated Ethernet Bundle Overview . . . . .	269
	Applying Layer 2 Policers to Gigabit Ethernet Interfaces . . . . .	270
	Examples: Applying Layer 2 Policers to a Gigabit Ethernet Interface . . . . .	270
	Examples: Configuring CoS on Aggregated Interfaces . . . . .	271
	Example: Configuring Scheduling Modes on Aggregated Interfaces . . . . .	273
<b>Chapter 17</b>	<b>Configuring Class of Service on 10-Gigabit Ethernet LAN/WAN PICs with SFP+ . . . . .</b>	<b>279</b>
	CoS on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview . . . . .	279
	DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC with SFP+ . . . . .	280
	BA and Fixed Classification on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview . . . . .	283
	Queuing on 10-Gigabit Ethernet LAN/WAN PICs Properties . . . . .	284
	Scheduling and Shaping on 10-Gigabit Ethernet LAN/WAN PICs Overview . . . . .	285
	Example: Configuring IEEE 802.1p BA Classifier on 10-Gigabit Ethernet LAN/WAN PICs . . . . .	286
	Mapping Forwarding Classes to CoS Queues on 10-Gigabit Ethernet LAN/WAN PICs . . . . .	287
	Example: Configuring Shaping Overhead on 10-Gigabit Ethernet LAN/WAN PICs . . . . .	288
	Configuring DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC . . . . .	288
<b>Chapter 18</b>	<b>Configuring Class of Service on MICs, MPCs, and MLCs . . . . .</b>	<b>291</b>
	CoS Features and Limitations on MIC and MPC Interfaces . . . . .	291
	Understanding IEEE 802.1p Inheritance push and swap from a Transparent Tag . . . . .	293
	Configuring IEEE 802.1p Inheritance push and swap from the Transparent Tag . . . . .	294
	Excess Bandwidth Distribution on MIC and MPC Interfaces Overview . . . . .	296
	Managing Dedicated and Remaining Queues for Static CoS Configurations on MIC and MPC Interfaces . . . . .	296
	Configuring the Maximum Number of Queues for MIC and MPC Interfaces . . . . .	296
	Configuring Remaining Common Queues on MIC and MPC Interfaces . . . . .	297
	Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs . . . . .	298
	CoS on Ethernet Pseudowires in Universal Edge Networks Overview . . . . .	300
	CoS on Application Services Modular Line Card Overview . . . . .	300
	CoS Implementation in HTTP Reverse Proxy Scenario . . . . .	301
	CoS Implementation in Transparent Proxy Scenario . . . . .	302
	CoS Implementation in Mixed-Mode Scenario . . . . .	302
	Configuring CoS on an Ethernet Pseudowire for Multiservice Edge Networks . . . . .	302

	Configuring a CoS Scheduling Policy on Logical Tunnel Interfaces . . . . .	304
	Verifying the Number of Dedicated Queues Configured on MIC and MPC Interfaces . . . . .	305
<b>Part 5</b>	<b>Configuration Statements</b>	
<b>Chapter 19</b>	<b>Configuration Statement Hierarchies . . . . .</b>	<b>309</b>
	[edit chassis] Hierarchy Level . . . . .	309
	[edit class-of-service] Hierarchy Level . . . . .	317
	[edit firewall] Hierarchy Level . . . . .	321
	Common Firewall Actions . . . . .	321
	Common IP Firewall Actions . . . . .	322
	Common IPv4 and IPv6 Firewall Actions . . . . .	322
	Common IP Firewall Match Conditions . . . . .	323
	Common IPv4 Firewall Match Conditions . . . . .	324
	Common Layer 2 Firewall Match Conditions . . . . .	324
	Complete [edit firewall] Hierarchy . . . . .	326
	[edit interfaces] Hierarchy Level . . . . .	333
<b>Chapter 20</b>	<b>Configuration Statements: Aggregated Ethernet Interfaces . . . . .</b>	<b>345</b>
	[edit class-of-service] Hierarchy Level . . . . .	345
	buffer-size (Schedulers) . . . . .	350
	drop-profile (Schedulers) . . . . .	351
	drop-profile-map (Schedulers) . . . . .	351
	excess-priority . . . . .	352
	excess-rate . . . . .	353
	forwarding-class (Interfaces) . . . . .	354
	interfaces . . . . .	355
	loss-priority (Scheduler Drop Profiles) . . . . .	357
	priority (Schedulers) . . . . .	358
	protocol (Schedulers) . . . . .	359
	scheduler-map (Interfaces and Traffic-Control Profiles) . . . . .	360
	scheduler-maps (For Most Interface Types) . . . . .	360
	schedulers (CoS) . . . . .	361
	transmit-rate (Schedulers) . . . . .	362
	unit . . . . .	364
<b>Chapter 21</b>	<b>Configuration Statements: BA Classifiers . . . . .</b>	<b>365</b>
	[edit class-of-service] Hierarchy Level . . . . .	365
	classifiers (Logical Interface) . . . . .	370
	classifiers (Routing Instance) . . . . .	371
	classifiers (Definition) . . . . .	372
	classifiers (Physical Interface) . . . . .	373
	code-points . . . . .	374
	copy-plp-all . . . . .	374
	dscp (AS PIC Classifiers) . . . . .	375
	dscp (CoS Interfaces) . . . . .	375
	dscp-ipv6 (CoS Rewrite Rules) . . . . .	376
	exp . . . . .	377
	forwarding-class (BA Classifiers) . . . . .	378

	ieee-802.1 (Rewrite Rules on Logical Interface) . . . . .	379
	ieee-802.1 (Classifier on Physical Interface) . . . . .	380
	ieee-802.1ad . . . . .	381
	import (Classifiers) . . . . .	382
	inet-precedence (CoS Rewrite Rules) . . . . .	382
	inet-precedence (Classifier on Physical Interface) . . . . .	383
	interfaces . . . . .	384
	loss-priority (BA Classifiers) . . . . .	386
	routing-instances (CoS) . . . . .	387
	system-defaults . . . . .	388
	unit . . . . .	389
<b>Chapter 22</b>	<b>Configuration Statements: Code Point Aliases . . . . .</b>	<b>391</b>
	[edit class-of-service] Hierarchy Level . . . . .	391
	code-point-aliases . . . . .	395
	dscp (Rewrite Rules) . . . . .	396
	dscp-ipv6 (CoS Rewrite Rules) . . . . .	397
	exp . . . . .	398
	ieee-802.1 (Rewrite Rules on Logical Interface) . . . . .	399
	inet-precedence (CoS Rewrite Rules) . . . . .	400
<b>Chapter 23</b>	<b>Configuration Statements: Forwarding Policy Options . . . . .</b>	<b>401</b>
	[edit class-of-service] Hierarchy Level . . . . .	401
	class (Forwarding Classes) . . . . .	406
	forwarding-class (Interfaces) . . . . .	406
	forwarding-class (Restricted Queues) . . . . .	407
	forwarding-classes . . . . .	408
	forwarding-classes-interface-specific . . . . .	409
	interfaces . . . . .	410
	output-forwarding-class-map . . . . .	411
	priority (Fabric Priority) . . . . .	412
	queue (Global Queues) . . . . .	413
	queue (Restricted Queues) . . . . .	413
	restricted-queues . . . . .	414
	unit . . . . .	415
<b>Chapter 24</b>	<b>Configuration Statements: MIC and MPC Interfaces . . . . .</b>	<b>417</b>
	[edit class-of-service] Hierarchy Level . . . . .	417
	adjust-minimum . . . . .	422
	adjust-percent . . . . .	422
	excess-priority . . . . .	423
	excess-rate . . . . .	424
	excess-rate-high . . . . .	425
	excess-rate-low . . . . .	425
	schedulers (CoS) . . . . .	426
	shaping-rate-excess-high . . . . .	427
	shaping-rate-excess-low . . . . .	428
	shaping-rate-priority-high . . . . .	429
	shaping-rate-priority-low . . . . .	430
	shaping-rate-priority-medium . . . . .	431

	traffic-control-profiles . . . . .	432
<b>Chapter 25</b>	<b>Configuration Statements: Packets Using Multifield Classifiers . . . . .</b>	<b>433</b>
	[edit class-of-service] Hierarchy Level . . . . .	433
	transparent . . . . .	437
	[edit firewall] Hierarchy Level . . . . .	437
	Common Firewall Actions . . . . .	438
	Common IP Firewall Actions . . . . .	438
	Common IPv4 and IPv6 Firewall Actions . . . . .	439
	Common IP Firewall Match Conditions . . . . .	439
	Common IPv4 Firewall Match Conditions . . . . .	440
	Common Layer 2 Firewall Match Conditions . . . . .	440
	Complete [edit firewall] Hierarchy . . . . .	442
	dscp (Multifield Classifier) . . . . .	449
	family (Multifield Classifier) . . . . .	450
	filter (Configuring) . . . . .	451
	forwarding-class (Multifield Classifiers) . . . . .	452
	from . . . . .	452
	loss-priority (Firewall Filter) . . . . .	453
	loss-priority (Simple Firewall Filter) . . . . .	453
	term (Simple Filter) . . . . .	454
	then (Services CoS) . . . . .	455
	[edit interfaces] Hierarchy Level . . . . .	455
	filter (Applying to an Interface) . . . . .	467
	simple-filter (Applying to an Interface) . . . . .	468
<b>Chapter 26</b>	<b>Configuration Statements: RED Drop Profiles . . . . .</b>	<b>469</b>
	[edit class-of-service] Hierarchy Level . . . . .	469
	drop-probability (Interpolated Value) . . . . .	473
	drop-profiles . . . . .	474
	fill-level (Interpolated Value) . . . . .	475
	fill-level (Drop Profiles) . . . . .	476
	interpolate . . . . .	476
<b>Chapter 27</b>	<b>Configuration Statements: Rewriting Packet Header Information . . . . .</b>	<b>477</b>
	[edit class-of-service] Hierarchy Level . . . . .	478
	code-point . . . . .	482
	default (CoS Host Outbound Traffic) . . . . .	482
	dscp (Rewrite Rules) . . . . .	483
	dscp (Rewrite Rules on Physical Interface) . . . . .	484
	dscp-ipv6 (CoS Rewrite Rules) . . . . .	485
	exp . . . . .	486
	exp-push-push-push . . . . .	487
	exp-swap-push-push . . . . .	488
	forwarding-class (BA Classifiers) . . . . .	489
	frame-relay-de (Defining Loss Priority Maps) . . . . .	490
	host-outbound-traffic (Class-of-Service) . . . . .	491
	ieee-802.1 (Rewrite Rules on Logical Interface) . . . . .	492
	ieee-802.1 (Host Outbound Traffic) . . . . .	493
	ieee-802.1 (Rewrite Rules on Physical Interface) . . . . .	493

	ieee-802.1ad . . . . .	494
	import (Rewrite Rules) . . . . .	495
	inet-precedence (CoS Rewrite Rules) . . . . .	495
	inet-precedence (Rewrite Rules on Physical Interface) . . . . .	496
	interfaces . . . . .	497
	loss-priority (BA Classifiers) . . . . .	499
	loss-priority-maps . . . . .	500
	loss-priority-maps (Assigning to an Interface) . . . . .	501
	protocol (Rewrite Rules) . . . . .	502
	rewrite-rules (CoS Host Outbound Traffic) . . . . .	503
	rewrite-rules (Definition) . . . . .	504
	rewrite-rules (Interfaces) . . . . .	505
	rewrite-rules (Physical Interfaces) . . . . .	506
	unit . . . . .	507
	vlan-tag . . . . .	508
<b>Chapter 28</b>	<b>Configuration Statements: Routing Engine Protocol Queue Assignments . . . . .</b>	<b>509</b>
	[edit class-of-service] Hierarchy Level . . . . .	509
	classifiers (Logical Interface) . . . . .	514
	dscp (Rewrite Rules) . . . . .	515
	dscp-code-point (CoS Host Outbound Traffic) . . . . .	516
	dscp-ipv6 (CoS Rewrite Rules) . . . . .	517
	exp . . . . .	518
	forwarding-class (Forwarding Policy) . . . . .	519
	host-outbound-traffic (Class-of-Service) . . . . .	520
	ieee-802.1 (Rewrite Rules on Logical Interface) . . . . .	521
	inet-precedence (CoS Rewrite Rules) . . . . .	522
	irb . . . . .	523
	protocol (Rewrite Rules) . . . . .	524
	rewrite-rules (Interfaces) . . . . .	525
	unit . . . . .	527
	vlan-tag . . . . .	528
<b>Chapter 29</b>	<b>Configuration Statements: Schedulers . . . . .</b>	<b>529</b>
	[edit class-of-service] Hierarchy Level . . . . .	530
	buffer-size (Schedulers) . . . . .	534
	delay-buffer-rate . . . . .	535
	drop-profile-map (Schedulers) . . . . .	536
	excess-priority . . . . .	537
	excess-rate . . . . .	538
	fabric (Class-of-Service) . . . . .	539
	forwarding-class (Interfaces) . . . . .	539
	guaranteed-rate . . . . .	540
	interfaces . . . . .	541
	loss-priority (Scheduler Drop Profiles) . . . . .	543
	output-traffic-control-profile . . . . .	544
	priority (Fabric Queues, Schedulers) . . . . .	545
	priority (Schedulers) . . . . .	546
	protocol (Schedulers) . . . . .	547

	scheduler (Fabric Queues) . . . . .	548
	scheduler (Scheduler Map) . . . . .	548
	scheduler-map (Fabric Queues) . . . . .	549
	scheduler-map (Interfaces and Traffic-Control Profiles) . . . . .	549
	scheduler-map-chassis . . . . .	550
	scheduler-maps (For Most Interface Types) . . . . .	551
	schedulers (CoS) . . . . .	552
	shaping-rate (Applying to an Interface) . . . . .	553
	shaping-rate (Oversubscribing an Interface) . . . . .	555
	traffic-control-profiles . . . . .	556
	transmit-rate (Schedulers) . . . . .	557
	unit . . . . .	559
	[edit interfaces] Hierarchy Level . . . . .	559
	schedulers (Interfaces) . . . . .	570
<b>Chapter 30</b>	<b>Configuration Statements: Tricolor Marking Policers . . . . .</b>	<b>571</b>
	[edit class-of-service] Hierarchy Level . . . . .	572
	classifiers (Definition) . . . . .	576
	code-points . . . . .	577
	drop-profile (Schedulers) . . . . .	577
	drop-profile-map (Schedulers) . . . . .	578
	dscp (Multifield Classifier) . . . . .	579
	dscp (Rewrite Rules) . . . . .	580
	dscp-ipv6 (CoS Rewrite Rules) . . . . .	581
	exp . . . . .	582
	forwarding-class (BA Classifiers) . . . . .	583
	ieee-802.1 (Rewrite Rules on Logical Interface) . . . . .	584
	import (Classifiers) . . . . .	585
	import (Rewrite Rules) . . . . .	585
	inet-precedence (CoS Rewrite Rules) . . . . .	586
	loss-priority (Scheduler Drop Profiles) . . . . .	587
	protocol (Schedulers) . . . . .	588
	rewrite-rules (Definition) . . . . .	589
	schedulers (CoS) . . . . .	590
	tri-color . . . . .	591
	[edit firewall] Hierarchy Level . . . . .	591
	Common Firewall Actions . . . . .	591
	Common IP Firewall Actions . . . . .	592
	Common IPv4 and IPv6 Firewall Actions . . . . .	592
	Common IP Firewall Match Conditions . . . . .	593
	Common IPv4 Firewall Match Conditions . . . . .	594
	Common Layer 2 Firewall Match Conditions . . . . .	594
	Complete [edit firewall] Hierarchy . . . . .	596
	action . . . . .	603
	family (Multifield Classifier) . . . . .	604
	filter (Configuring) . . . . .	605
	logical-interface-policer . . . . .	606
	loss-priority (Firewall Filter) . . . . .	607
	loss-priority (Simple Firewall Filter) . . . . .	607

	policer (Configuring) . . . . .	608
	shared-bandwidth-policer (Configuring) . . . . .	609
	term (Normal Filter) . . . . .	610
	then (Services CoS) . . . . .	611
	three-color-policer (Applying) . . . . .	612
	three-color-policer (Configuring) . . . . .	613
	[edit interfaces] Hierarchy Level . . . . .	614
	filter (Applying to an Interface) . . . . .	625
	input-policer . . . . .	626
	input-three-color . . . . .	627
	layer2-policer . . . . .	628
	output-policer . . . . .	629
	output-three-color . . . . .	630
<b>Chapter 31</b>	<b>Configuration Statements: Tunnels CoS . . . . .</b>	<b>631</b>
	[edit class-of-service] Hierarchy Level . . . . .	631
	code-point . . . . .	635
	dscp (Rewrite Rules) . . . . .	636
	dscp-ipv6 (CoS Rewrite Rules) . . . . .	637
	exp . . . . .	638
	exp-push-push-push . . . . .	639
	exp-swap-push-push . . . . .	640
	forwarding-class (BA Classifiers) . . . . .	641
	ieee-802.1 (Rewrite Rules on Logical Interface) . . . . .	642
	import (Rewrite Rules) . . . . .	643
	inet-precedence (CoS Rewrite Rules) . . . . .	643
	interfaces . . . . .	644
	loss-priority (BA Classifiers) . . . . .	646
	protocol (Rewrite Rules) . . . . .	647
	rewrite-rules (Definition) . . . . .	648
	rewrite-rules (Interfaces) . . . . .	649
	unit . . . . .	651
	[edit interfaces] Hierarchy Level . . . . .	651
	copy-tos-to-outer-ip-header . . . . .	663



# List of Figures

<b>Part 1</b>	<b>Overview</b>	
<b>Chapter 1</b>	<b>Understanding How Class of Service Alleviates Congestion and Defines Traffic Forwarding Behavior</b>	<b>3</b>
	Figure 1: Packet Flow Through CoS-Configurable Components	6
	Figure 2: Packet Flow Through CoS-Configurable Components	9
<b>Chapter 2</b>	<b>Understanding Class of Service Packet Flow Across a Network</b>	<b>15</b>
	Figure 3: Packet Flow Across the Network	16
	Figure 4: CoS Classifier, Queues, and Scheduler	17
	Figure 5: Packet Flow Through CoS-Configurable Components	17
<b>Part 2</b>	<b>Configuring Class of Service</b>	
<b>Chapter 3</b>	<b>Assigning Service Levels to Packets Using Behavior Aggregate Classifiers</b>	<b>21</b>
	Figure 6: How a Classifier Works	26
<b>Chapter 4</b>	<b>Assigning Service Levels to Packets Using Multifield Classifiers</b>	<b>69</b>
	Figure 7: How a Classifier Works	70
<b>Chapter 5</b>	<b>Controlling Access to the Network Using Traffic Policing</b>	<b>79</b>
	Figure 8: Network Traffic and Burst Rates	81
	Figure 9: Flow of Tricolor Marking Policer Operation	90
<b>Chapter 6</b>	<b>Defining Forwarding Behavior Based on Forwarding Classes</b>	<b>111</b>
	Figure 10: Customer-Facing and Core-Facing Forwarding Classes	118
	Figure 11: Sample CoS-Based Forwarding	130
<b>Chapter 10</b>	<b>Controlling Congestion Using Scheduler RED Drop Profiles and Buffers</b>	<b>199</b>
	Figure 12: Segmented and Interpolated Drop Profiles	201
	Figure 13: Segmented and Interpolated Drop Profiles	205
<b>Chapter 11</b>	<b>Altering Outgoing Packet Headers Using Rewrite Rules to Ensure Forwarding Behavior</b>	<b>211</b>
	Figure 14: Packet Flow Across the Network	212
<b>Part 4</b>	<b>Configuring Line Card-Specific and Interface-Specific Functionality</b>	
<b>Chapter 14</b>	<b>Configuring Class of Service for Tunnels</b>	<b>247</b>
	Figure 15: CoS with a Tunnel Configuration	249
<b>Chapter 15</b>	<b>Configuring Class of Service on IQ and Enhanced IQ (IQE) PICs</b>	<b>253</b>

Chapter 16

Figure 16: Topology to Verify Link Redundancy Support for L2TP LNS CoS . . . . 259

**Configuring Class of Service on Aggregated, Channelized, and Gigabit Ethernet Interfaces . . . . . 263**

Figure 17: Scaled Mode for Aggregated Ethernet Interfaces . . . . . 276

Figure 18: Replicated Mode for Aggregated Ethernet Interfaces . . . . . 278

# List of Tables

	<b>About the Documentation</b> . . . . .	<b>xxi</b>
	Table 1: Notice Icons . . . . .	xxiii
	Table 2: Text and Syntax Conventions . . . . .	xxiii
<b>Part 1</b>	<b>Overview</b>	
<b>Chapter 1</b>	<b>Understanding How Class of Service Alleviates Congestion and Defines Traffic Forwarding Behavior</b> . . . . .	<b>3</b>
	Table 3: CoS Mappings—Inputs and Outputs . . . . .	9
<b>Part 2</b>	<b>Configuring Class of Service</b>	
<b>Chapter 3</b>	<b>Assigning Service Levels to Packets Using Behavior Aggregate Classifiers</b> . . . . .	<b>21</b>
	Table 4: Default CoS Value Aliases . . . . .	22
	Table 5: Logical Interface Classifier Combinations . . . . .	32
	Table 6: Default IP Precedence (ipprec-compatibility) Classifier . . . . .	39
	Table 7: Default IP Precedence (ipprec-default) Classifier . . . . .	39
	Table 8: Default IP Precedence (ipprec-default) Classifier . . . . .	40
	Table 9: Default VPLS Classifiers . . . . .	41
	Table 10: Default DSCP and DSCP IPv6 Classifiers . . . . .	43
	Table 11: Default MPLS EXP Classification . . . . .	48
	Table 12: Default MPLS Classifier . . . . .	50
	Table 13: Default MPLS EXP Classifier . . . . .	50
	Table 14: Default IEEE 802.1p Classifier . . . . .	56
	Table 15: Default MPLS EXP Classifier . . . . .	58
	Table 16: Default IEEE 802.1ad Classifier . . . . .	67
<b>Chapter 5</b>	<b>Controlling Access to the Network Using Traffic Policing</b> . . . . .	<b>79</b>
	Table 17: Policer Actions . . . . .	82
	Table 18: Color-Blind Mode TCM Color-to-PLP Mapping . . . . .	93
	Table 19: Color-Aware Mode TCM PLP Mapping . . . . .	94
	Table 20: Color-Blind Mode TCM Color-to-PLP Mapping . . . . .	96
	Table 21: Color-Aware Mode TCM Mapping . . . . .	97
	Table 22: Devices Versus TCM . . . . .	101
	Table 23: Tricolor Marking Policer Statements . . . . .	103
<b>Chapter 6</b>	<b>Defining Forwarding Behavior Based on Forwarding Classes</b> . . . . .	<b>111</b>
	Table 24: Default Forwarding Classes . . . . .	114
	Table 25: Sample Forwarding Class-to-Queue Mapping . . . . .	118
	Table 26: Routing Engine Protocol Queue Assignments . . . . .	135

<b>Chapter 7</b>	<b>Defining Output Queue Properties Using Schedulers . . . . .</b>	<b>143</b>
	Table 27: Buffer Size Temporal Value Ranges by Routing Device Type . . . . .	147
	Table 28: Recommended Delay Buffer Sizes . . . . .	149
	Table 29: Maximum Delay Buffer with q-pic-large-buffer Enabled by Interface . . . . .	149
	Table 30: Delay-Buffer Calculations . . . . .	151
	Table 31: NxDSO Transmission Rates and Delay Buffers . . . . .	152
<b>Chapter 8</b>	<b>Controlling Bandwidth Using Scheduler Rates . . . . .</b>	<b>163</b>
	Table 32: Bandwidth and Delay Buffer Allocations by Configuration Scenario . .	167
	Table 33: Bandwidth and Delay Buffer Allocations by Configuration Scenario . .	177
	Table 34: Fine-Grained Queuing and Scheduling Support by Interface or PIC Type . . . . .	183
	Table 35: Fine-Grained Queuing and Scheduling Support by MIC or MPC Type . . . . .	185
<b>Chapter 11</b>	<b>Altering Outgoing Packet Headers Using Rewrite Rules to Ensure Forwarding Behavior . . . . .</b>	<b>211</b>
	Table 36: Default Packet Header Rewrite Mappings . . . . .	214
	Table 37: Default MPLS EXP Rewrite Table . . . . .	224
<b>Part 4</b>	<b>Configuring Line Card-Specific and Interface-Specific Functionality</b>	
<b>Chapter 17</b>	<b>Configuring Class of Service on 10-Gigabit Ethernet LAN/WAN PICs with SFP+ . . . . .</b>	<b>279</b>
	Table 38: CoS Statements Supported on the 10-Gigabit Ethernet LAN/WAN PICs . . . . .	280
	Table 39: Port Groups on 10-Gigabit Ethernet LAN/WAN PICs . . . . .	285
<b>Chapter 18</b>	<b>Configuring Class of Service on MICs, MPCs, and MLCs . . . . .</b>	<b>291</b>
	Table 40: CoS Limitations on MIC and MPC Interfaces . . . . .	291

# About the Documentation

- Documentation and Release Notes on page xxi
- Supported Platforms on page xxi
- Using the Examples in This Manual on page xxi
- Documentation Conventions on page xxiii
- Documentation Feedback on page xxv
- Requesting Technical Support on page xxv

## Documentation and Release Notes

---

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

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

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

## Supported Platforms

---

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

- EX Series

## Using the Examples in This Manual

---

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

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

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

## Merging a Full Example

To merge a full example, follow these steps:

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

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

```
system {
  scripts {
    commit {
      file ex-script.xml;
    }
  }
}
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.xml; }
```

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

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

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

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

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

## Documentation Conventions

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

Table 2: Text and Syntax Conventions

Convention	Description	Examples
<b>Bold text like this</b>	Represents text that you type.	To enter configuration mode, type the <b>configure</b> command:  user@host> <b>configure</b>

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

Convention	Description	Examples
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> <b>show chassis alarms</b>  No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"><li>Introduces or emphasizes important new terms.</li><li>Identifies guide names.</li><li>Identifies RFC and Internet draft titles.</li></ul>	<ul style="list-style-type: none"><li>A policy <i>term</i> is a named structure that defines match conditions and actions.</li><li><i>Junos OS CLI User Guide</i></li><li>RFC 1997, <i>BGP Communities Attribute</i></li></ul>
<i>Italic text like this</i>	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name:  [edit] root@# <b>set system domain-name domain-name</b>
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	<ul style="list-style-type: none"><li>To configure a stub area, include the <b>stub</b> statement at the [edit protocols ospf area area-id] hierarchy level.</li><li>The console port is labeled <b>CONSOLE</b>.</li></ul>
< > (angle brackets)	Encloses optional keywords or variables.	<b>stub &lt;default-metric metric&gt;;</b>
(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.	<b>broadcast   multicast</b>  <b>(string1   string2   string3)</b>
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	<b>rsvp { # Required for dynamic MPLS only</b>
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	<b>community name members [ community-ids ]</b>
Indentation and braces ( { } )	Identifies a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop address; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		
Bold text like this	Represents graphical user interface (GUI) items you click or select.	<ul style="list-style-type: none"><li>In the Logical Interfaces box, select <b>All Interfaces</b>.</li><li>To cancel the configuration, click <b>Cancel</b>.</li></ul>

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

Convention	Description	Examples
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select <b>Protocols&gt;Ospf</b> .

## Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page of the Juniper Networks TechLibrary site at <http://www.juniper.net/techpubs/index.html>, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at <http://www.juniper.net/techpubs/feedback/>.
- E-mail—Send your comments to [techpubs-comments@juniper.net](mailto: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 J-Care or Partner Support Service support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

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

## Self-Help Online Tools and Resources

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

- Find CSC offerings: <http://www.juniper.net/customers/support/>
- Search for known bugs: <http://www2.juniper.net/kb/>
- Find product documentation: <http://www.juniper.net/techpubs/>
- Find solutions and answer questions using our Knowledge Base: <http://kb.juniper.net/>

- Download the latest versions of software and review release notes:  
<http://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications:  
<http://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum:  
<http://www.juniper.net/company/communities/>
- Open a case online in the CSC Case Management tool: <http://www.juniper.net/cm/>

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

## Opening a Case with JTAC

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

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

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

## PART 1

# Overview

- [Understanding How Class of Service Alleviates Congestion and Defines Traffic Forwarding Behavior on page 3](#)
- [Understanding Class of Service Packet Flow Across a Network on page 15](#)



## CHAPTER 1

# Understanding How Class of Service Alleviates Congestion and Defines Traffic Forwarding Behavior

- [Understanding How Class of Service Manages Congestion and Controls Service Levels in the Network on page 3](#)
- [The Junos OS CoS Components Used to Manage Congestion and Control Service Levels on page 5](#)
- [Mapping CoS Component Inputs to Outputs on page 9](#)
- [Default Junos OS CoS Settings on page 12](#)

## 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 divide traffic into 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.

## 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.

## 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 5](#)

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

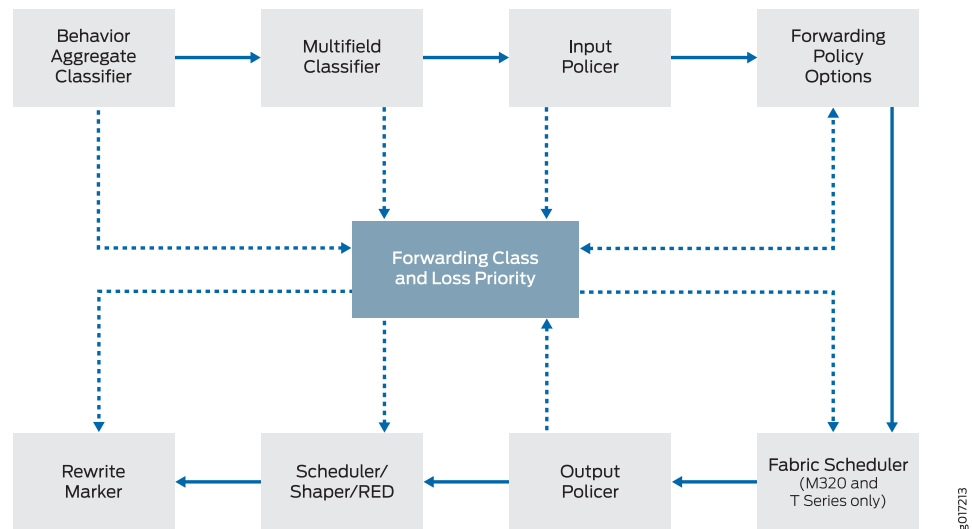
Any CoS implementation must work consistently end to end through the network. A standards-based, vendor-neutral CoS implementation satisfies this requirement best. Junos OS CoS features interoperate with other vendors' CoS implementations because they are based on IETF Differentiated Services (DiffServ) 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 1 on page 6 shows the components of the Junos OS CoS features, illustrating the sequence in which they interact.

**Figure 1: 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 26 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.

See [“Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields” on page 69](#) 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 3D Universal Edge Routers, 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 [“Forwarding Classes Overview” on page 111](#) 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 139](#) 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 126](#) 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 [“Schedulers Overview” on page 143](#) for more information on schedulers.

- **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 79](#) 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 212](#) for more information on rewrite rules.

**Related  
Documentation**

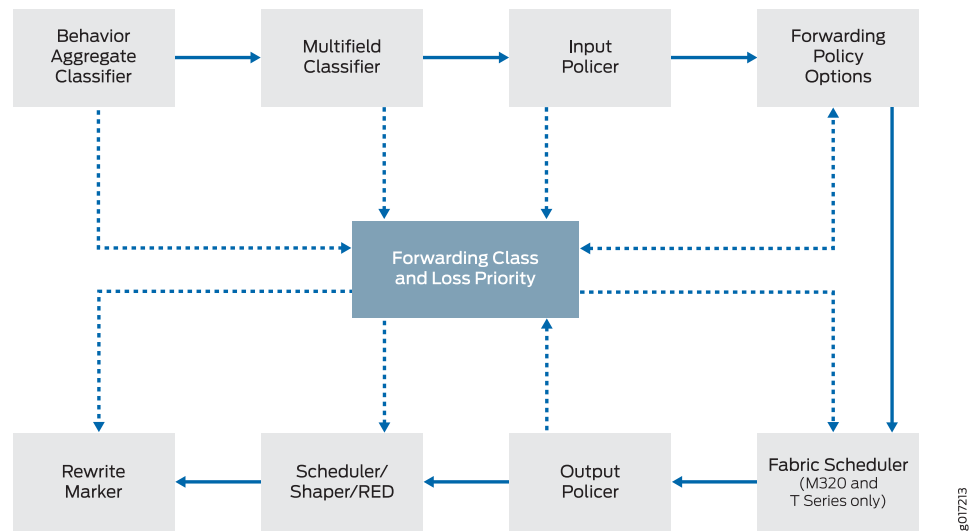
- [Understanding How Class of Service Manages Congestion and Controls Service Levels in the Network on page 3](#)

## 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 1 on page 6 shows the components of the Junos OS CoS features, illustrating the sequence in which they interact.

**Figure 2: 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 9.

**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.

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

CoS Mappings	Inputs	Outputs	Comments
<code>rewrite-rules</code>	<code>forwarding-class</code> <code>loss-priority</code>	<code>code-points</code>	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.

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 36](#) 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 199](#) 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 157](#) for more information on mapping forwarding classes to schedulers.

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 215 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 26](#)
- [Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199](#)
- [Configuring Scheduler Maps on page 157](#)
- [Applying Default Rewrite Rules](#)

---

## 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.

---

#### 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

```

### 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](#)
- [Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26](#)
- [Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 200](#)
- [Default Schedulers Overview on page 145](#)
- [Forwarding Classes and Fabric Priority Queues on page 161](#)

## CHAPTER 2

# Understanding Class of Service Packet Flow Across a Network

- [How CoS Applies to Packet Flow Across a Network on page 15](#)
- [Packet Flow Through the Junos OS CoS Process Overview on page 16](#)

### How CoS Applies to Packet Flow Across a Network

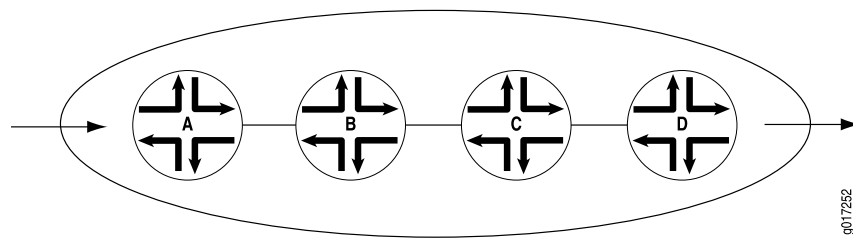
---

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 3 on page 16](#), 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.

Figure 3: Packet Flow Across the Network



- Related Documentation**
- [The Junos OS CoS Components Used to Manage Congestion and Control Service Levels on page 5](#)

## 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*.

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).

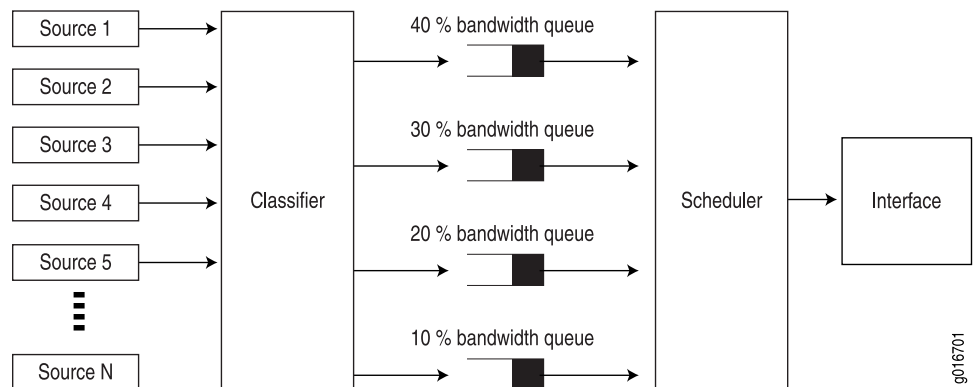
At the **[edit class-of-service interfaces]** hierarchy level, the **scheduler-map** and **rewrite-rules** statements affect the outgoing packets, and the **classifiers** statement affects the incoming packets.

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).

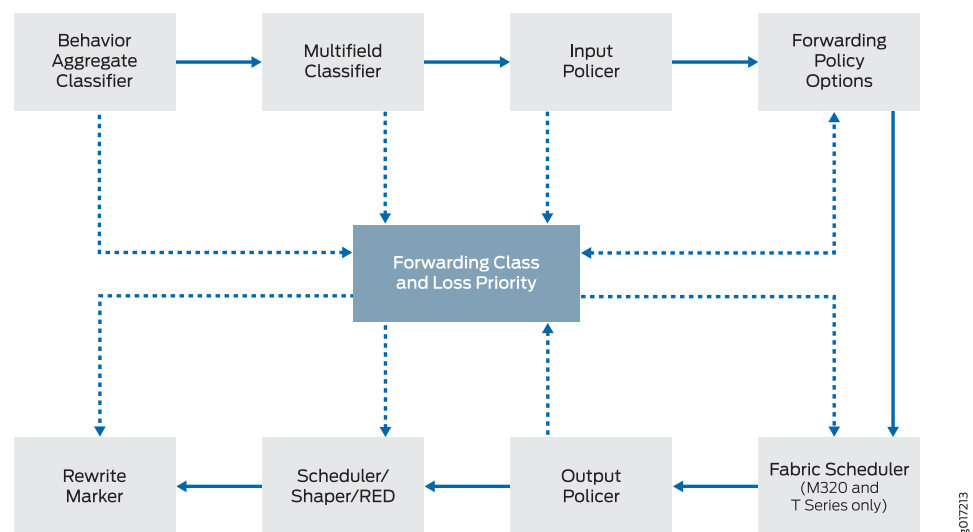
7. The scheduler map assigns a scheduler to each forwarding class (at the **[edit class-of-service scheduler-maps]** hierarchy level).
8. The drop-profile defines how aggressively to drop packets that are using a particular scheduler (at the **[edit class-of-service drop-profiles]** hierarchy level).
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.

Figure 4 on page 17 and Figure 5 on page 17 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 17 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.

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*
- *Packet Flow on Juniper Networks M Series Multiservice Edge Routers*
- *Packet Flow on MX Series 3D Universal Edge Routers*
- *Packet Flow on Juniper Networks T Series Core Routers*

## PART 2

# Configuring Class of Service

- [Assigning Service Levels to Packets Using Behavior Aggregate Classifiers on page 21](#)
- [Assigning Service Levels to Packets Using Multifield Classifiers on page 69](#)
- [Controlling Access to the Network Using Traffic Policing on page 79](#)
- [Defining Forwarding Behavior Based on Forwarding Classes on page 111](#)
- [Defining Output Queue Properties Using Schedulers on page 143](#)
- [Controlling Bandwidth Using Scheduler Rates on page 163](#)
- [Setting Transmission Order Using Scheduler Priorities on page 195](#)
- [Controlling Congestion Using Scheduler RED Drop Profiles and Buffers on page 199](#)
- [Altering Outgoing Packet Headers Using Rewrite Rules to Ensure Forwarding Behavior on page 211](#)



## CHAPTER 3

# Assigning Service Levels to Packets Using Behavior Aggregate Classifiers

- [Default Aliases for CoS Value Bit Patterns Overview on page 22](#)
- [Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26](#)
- [Default Behavior Aggregate Classification Overview on page 29](#)
- [Defining Aliases for CoS Value Bit Patterns on page 30](#)
- [Applying Behavior Aggregate Classifiers to Logical Interfaces on page 32](#)
- [Using BA Classifiers to Set PLP on page 35](#)
- [Configuring Behavior Aggregate Classifiers on page 36](#)
- [Default IP Precedence Classifier on page 38](#)
- [Default IP Precedence Classifier \(ipprec-default\) on page 40](#)
- [Understanding DSCP Classification for VPLS on page 41](#)
- [VPLS and Default CoS Classification on page 41](#)
- [Default DSCP and DSCP IPv6 Classifiers on page 42](#)
- [Applying DSCP Classifiers to MPLS Traffic on page 43](#)
- [Default MPLS EXP Classifier on page 47](#)
- [Configuring CoS for MPLS Traffic on page 48](#)
- [Default MPLS EXP Classifier on page 49](#)
- [Applying MPLS EXP Classifiers to Routing Instances on page 50](#)
- [Applying MPLS EXP Classifiers for Explicit-Null Labels on page 55](#)
- [Default IEEE 802.1p Classifier on page 56](#)
- [DSCP IPv6 Rewrites and Forwarding Class Maps on page 57](#)
- [Applying DSCP IPv6 Classifiers on page 58](#)
- [Applying MPLS EXP Classifiers to Routing Instances on page 58](#)
- [Applying MPLS EXP Classifiers for Explicit-Null Labels on page 63](#)
- [Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets on page 64](#)
- [Default IEEE 802.1ad Classifier on page 66](#)
- [Configuring and Applying IEEE 802.1ad Classifiers on page 67](#)

## 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 4 on page 22 shows the default mappings between the CoS values and standard aliases.

**Table 4: Default CoS Value Aliases**

Default CoS Value Alias	CoS Value
<b>DSCP and DSCP IPv6 CoS Aliases and CoS Values</b>	
<b>ef</b>	101110
<b>af11</b>	001010
<b>af12</b>	001100
<b>af13</b>	001110
<b>af21</b>	010010
<b>af22</b>	010100
<b>af23</b>	010110
<b>af31</b>	011010
<b>af32</b>	011100
<b>af33</b>	011110
<b>af41</b>	100010

Table 4: Default CoS Value Aliases (*continued*)

Default CoS Value Alias	CoS Value
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
ef1	011

Table 4: Default CoS Value Aliases (*continued*)

Default CoS Value Alias	CoS Value
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
be1	001

Table 4: Default CoS Value Aliases (*continued*)

Default CoS Value Alias	CoS Value
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 30](#)
- [Default IP Precedence Classifier on page 38](#)
- [Default DSCP and DSCP IPv6 Classifiers on page 42](#)
- [Default MPLS EXP Classifier on page 47](#)
- [Default IEEE 802.1p Classifier on page 56](#)
- [Default IEEE 802.1ad Classifier on page 66](#)
- [code-point-aliases on page 395](#)

## 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. 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.



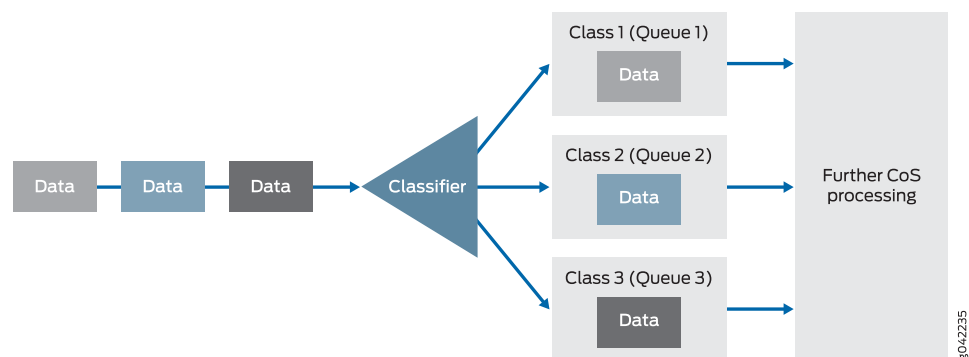
**NOTE:** In this document we generically refer to the various types of BA classification bits in the packet header as the CoS value.

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 69](#)) 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 212](#)), 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 discard (RED) algorithm to control packet discard during periods of congestion.

[Figure 6 on page 26](#) provides a high-level illustration of how a classifier works.

**Figure 6: How a Classifier Works**



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, 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 212](#).



**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 3D Universal Edge Routers, 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.



**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. For more information about multifield classifiers, see [“Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields” on page 69](#).

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 145](#)), is in effect regardless of any custom BA classifier or forwarding class definitions, until you configure a custom scheduler (see [“Configuring Schedulers” on page 146](#)).

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 47](#)) 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*).

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 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 56](#).



**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.

**Related  
Documentation**

- [Default IP Precedence Classifier on page 38](#)
- [Default DSCP and DSCP IPv6 Classifiers on page 42](#)
- [Default MPLS EXP Classifier on page 47](#)
- [Default IEEE 802.1p Classifier on page 56](#)
- [Default IEEE 802.1ad Classifier on page 66](#)
- [Configuring Behavior Aggregate Classifiers on page 36](#)
- [Applying Behavior Aggregate Classifiers to Logical Interfaces on page 32](#)
- [Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 69](#)
- [Rewriting Packet Headers to Ensure Forwarding Behavior on page 212](#)

## Default Behavior Aggregate Classification Overview

The software automatically assigns an implicit default IP precedence classifier to all logical interfaces.



**NOTE:** Only the IEEE 802.1p classifier is supported in Layer 2 interfaces. You must explicitly apply this classifier to the interface as shown in [“Default IEEE 802.1p Classifier” on page 56](#).

If you enable the MPLS protocol family on a logical interface, a default MPLS EXP classifier is automatically applied to that logical interface.

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



**NOTE:** Although several code points 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.

### Related Documentation

- [Default IP Precedence Classifier on page 38](#)
- [Default MPLS EXP Classifier on page 49](#)
- [Default DSCP and DSCP IPv6 Classifiers on page 42](#)
- [Default IEEE 802.1p Classifier on page 56](#)
- [Default IEEE 802.1ad Classifier on page 66](#)
- [Default IP Precedence Classifier \(ipprec-default\) on page 40](#)

## 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.
- **ieee-802.1ad**—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
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.
  - 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 22](#)
- [Applying Behavior Aggregate Classifiers to Logical Interfaces on page 32](#)

## Applying Behavior Aggregate Classifiers to Logical Interfaces

This topic describes how to apply BA classifiers to logical interfaces.

When you apply behavior aggregate (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 5 on page 32](#) shows the supported combinations. In this table, the OSE PICs refer to the 10-port 10-Gigabit OSE PICs.

**Table 5: 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
<b>dscp</b> and <b>inet-precedence</b>	No	No	No	No	No
<b>dscp-ipv6</b> and ( <b>dscp</b>   <b>inet-precedence</b> )	Yes	Yes	Yes	No	No
<b>exp</b> and <b>ieee 802.1</b>	Yes	Yes	No	No	No
<b>ieee 802.1</b> and ( <b>dscp</b>   <b>dscp-ipv6</b>   <b>exp</b>   <b>inet-precedence</b> )	Yes	Yes	No	No	Yes
<b>exp</b> and ( <b>dscp</b>   <b>dscp-ipv6</b>   <b>inet-precedence</b> )	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:** If an interface is mounted on an M Series router FPC, you can apply only the default exp classifier. If an interface is mounted on 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/L3VPN routing instances (LSI interfaces). The DSCP-based classification for MPLS packets for Layer 2 VPNs is not supported.



**NOTE:** If you do not apply a DSCP classifier, the default EXP classifier is applied to MPLS traffic.

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

- In a Layer 3 VPN (L3VPN) 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 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 (L3VPN) using a 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 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 routing-instance-name classifiers]** hierarchy level. For details, see [“Applying MPLS EXP Classifiers to Routing Instances” on page 50](#).
- 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** for the **fc-af11-class** forwarding class and a corresponding IPv6 DSCP classifier. 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-classifier {
    forwarding-class fc-af11-class {
      loss-priority low code-points af11;
    }
  }
}
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 43](#)

## Using BA Classifiers to Set PLP

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

```

[edit class-of-service]
classifiers {
    (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) classifier-name {
        import (classifier-name | default);
        forwarding-class class-name {

```

```
        loss-priority (low | medium-low | medium-high | high) code-points [ aliases ]  
        [ bit-patterns ];  
    }  
}
```

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

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

```
class-of-service {  
  classifiers {  
    dscp dscp-cl {  
      forwarding-class assured-forwarding {  
        loss-priority medium-low {  
          code-points 101110;  
        }  
      }  
    }  
  }  
}
```

To use this classifier, you must configure the settings for the **assured-forwarding** forwarding class at the **[edit class-of-service forwarding-classes queue *queue-number* assured-forwarding]** hierarchy level. For more information, see [“Forwarding Classes Overview” on page 111](#).

---

## 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 9](#).

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 139.

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

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 [“Forwarding Classes Overview” on page 111](#).

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 26](#)
- [Applying Behavior Aggregate Classifiers to Logical Interfaces on page 32](#)
- [Enabling Tricolor Marking and Limitations of Three-Color Policers on page 100](#)

---

## 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 6 on page 39](#).

**Table 6: 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 7 on page 39](#) shows the forwarding class and PLP that are assigned to the IP precedence bits when you apply the default IP precedence classifier.

**Table 7: 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

Table 7: Default IP Precedence (ipprec-default) Classifier (*continued*)

IP Precedence Bits	Forwarding Class	PLP
111	network-control	high

**Related Documentation**

- [Applying Behavior Aggregate Classifiers to Logical Interfaces on page 32](#)

## Default IP Precedence Classifier (ipprec-default)

There are two separate tables for default IP precedence classification. All logical interfaces are implicitly assigned the **ipprec-compatibility** classifier by default, as described in [Table 6 on page 39](#).

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 7 on page 39](#) shows the forwarding class and PLP that are assigned to the IP precedence CoS bits when you apply the default IP precedence classifier.

Table 8: Default IP Precedence (ipprec-default) Classifier

Code Point	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

## 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 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.

### Related Documentation

- [Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26](#)

## VPLS and Default CoS Classification

A VPLS routing instance with the **no-tunnel-services** option configured has a default 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, custom classifier for VPLS routing instances on T Series and LMNR based FPC for M320 is not supported. When a wild card configuration or an 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 3D Universal Edge Routers, and T Series Core Routers), the default classification applied to **no-tunnel-services** VPLS packets are shown in [Table 9 on page 41](#).

**Table 9: Default VPLS Classifiers**

MPLS Label EXP Bits	Forwarding Class/Queue
000	0

Table 9: Default VPLS Classifiers (*continued*)

MPLS Label EXP Bits	Forwarding Class/Queue
001	1
010	2
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 Forwarding Classes” on page 116](#).

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.

## 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 10 on page 43 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 10: 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 26](#)
- [Default Aliases for CoS Value Bit Patterns Overview on page 22](#)
- [Changing the Default Queuing and Marking of Host Outbound Traffic on page 137](#)
- [classifiers \(Logical Interface\) on page 370](#)

## 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 44](#)
- [Applying a DSCP Classifier to MPLS Traffic for L3VPN/VPLS on page 45](#)

## 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 1.1.1.1/24
user@host # set family iso
user@host # set family inet6 address 2000::1/64
user@host # set family mpls
```

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]
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 1.1.1.1/24;
}
family iso;
family inet6 {
    address 2000::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 ];
}
```

```
[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 dscp1]
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. Attach the classifier to the routing instance.

```
[edit class-of-service routing-instances vpls1]
user@host # set classifiers dscp dscp1
```



**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 ];
    }
    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 32](#)

## 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 11 on page 48](#) lists the default MPLS classifier mapping of EXP bits to forwarding classes and loss priorities..

**Table 11: 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

**Related Documentation**

- [Configuring Class of Service for MPLS LSPs](#)
- [Default Aliases for CoS Value Bit Patterns Overview on page 22](#)
- [code-point-aliases on page 395](#)

## Configuring CoS for MPLS Traffic

To configure class of service (CoS) for Multiprotocol Label Switching (MPLS) packets in a label-switched path (LSP), include the **class-of-service** statement with the appropriate CoS value:

```
class-of-service 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.

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*]
- [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 **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.

**Related Documentation**

- [Default MPLS EXP Classifier on page 47](#)
- *MPLS Applications Feature Guide for Routing Devices*

## Default MPLS EXP Classifier

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.

To configure code point aliases for MPLS EXP CoS markers, map alias names to bit patterns at the [edit class-of-service code-point-aliases exp] hierarchy level.

Table 12 on page 50 lists the default MPLS classifier mapping of EXP bits to forwarding classes and loss priorities.

**Table 12: Default MPLS Classifier**

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	high
110	network-control	low
111	network-control	high

**Related Documentation**

- [Default Aliases for CoS Value Bit Patterns Overview on page 22](#)
- [code-point-aliases on page 395](#)

## Applying MPLS EXP Classifiers to Routing Instances

This topic shows how to apply MPLS EXP classifiers to logical interfaces.

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 13 on page 50.

**Table 13: 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

Table 13: Default MPLS EXP Classifier (*continued*)

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

The following topics are included:

- [Configuring and Applying Custom MPLS EXP Classifiers to Routing Instances on page 51](#)
- [Applying Global Classifiers and Wildcard Routing Instances on page 52](#)
- [Example: Applying Global MPLS EXP Classifiers to Routing Instances on page 52](#)
- [Applying Classifiers by Using Wildcard Routing Instances on page 53](#)
- [Verifying the Classifiers Associated with Routing Instances on page 54](#)

## 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.

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

Complete the following steps to apply a custom classifier to the 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. 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.

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:** Wildcards 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\**, the 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 the 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.

## Example: 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#
user@host# set vpn1 vrf-table-label
user@host# set vpn2 vrf-table-label
user@host# set vpn3 vrf-table-label
```

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 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**.

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#
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 routing-instance **vpn-red**.

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

4. 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;
  }
}
```

## 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 command-line interface (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
```

- Related Documentation**
- [Configuring Behavior Aggregate Classifiers on page 36](#)
  - [Default MPLS EXP Classifier on page 47](#)
  - [Applying MPLS EXP Classifiers for Explicit-Null Labels on page 55](#)

## 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 for Routing Devices*.

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 32](#).

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]
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
```

#### Related Documentation

- [Configuring Behavior Aggregate Classifiers on page 36](#)
- [Default MPLS EXP Classifier on page 47](#)
- [Applying MPLS EXP Classifiers to Routing Instances on page 50](#)

## Default IEEE 802.1p Classifier

Table 14 on page 56 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 14: 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

Table 14: Default IEEE 802.1p Classifier (*continued*)

IEEE 802.1p CoS Value	Forwarding Class	PLP
101	assured-forwarding	high
110	network-control	low
111	network-control	high

**Related Documentation**

- [Applying Behavior Aggregate Classifiers to Logical Interfaces on page 32](#)
- [Default IEEE 802.1ad Classifier on page 66](#)

## 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 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 134](#)

## Applying DSCP IPv6 Classifiers

For M320 and T Series routers and EX Series switches, you can apply separate classifiers for IPv4 and IPv6 packets per logical interface by including the **classifiers** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level and specifying the **dscp** and **dscp-ipv6** classifier types:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
  classifiers dscp (classifier-name | default) family (mpls | inet);
  classifiers dscp-ipv6 (classifier-name | default) family (mpls | inet));
```

For M Series router enhanced FPCs, you cannot apply separate classifiers for IPv4 and IPv6 packets on a single logical 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 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 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.

For more information, see [“Applying Behavior Aggregate Classifiers to Logical Interfaces” on page 32](#).

## Applying MPLS EXP Classifiers to Routing Instances

This topic shows how to apply MPLS EXP classifiers to logical interfaces.

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 13 on page 50](#).

**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

Table 15: Default MPLS EXP Classifier (*continued*)

MPLS EXP Bits	Forwarding Class	Loss Priority
100	assured-forwarding	low
101	assured-forwarding	high
110	network-control	low
111	network-control	high

The following topics are included:

- [Configuring and Applying Custom MPLS EXP Classifiers to Routing Instances on page 59](#)
- [Applying Global Classifiers and Wildcard Routing Instances on page 60](#)
- [Example: Applying Global MPLS EXP Classifiers to Routing Instances on page 60](#)
- [Applying Classifiers by Using Wildcard Routing Instances on page 62](#)
- [Verifying the Classifiers Associated with Routing Instances on page 63](#)

## 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.

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

Complete the following steps to apply a custom classifier to the 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. 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.

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

- 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:** Wildcards 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*`, the 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 the 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.

## Example: 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#
user@host# set vpn1 vrf-table-label
user@host# set vpn2 vrf-table-label
user@host# set vpn3 vrf-table-label
```

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 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**.

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#
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 routing-instance **vpn-red**.

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

4. 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;
  }
}
```

## 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 command-line interface (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
```

**Related Documentation**

- [Configuring Behavior Aggregate Classifiers on page 36](#)
- [Default MPLS EXP Classifier on page 47](#)
- [Applying MPLS EXP Classifiers for Explicit-Null Labels on page 55](#)

## 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 for Routing Devices*.

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 32](#).

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]
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
```

**Related  
Documentation**

- [Configuring Behavior Aggregate Classifiers on page 36](#)
- [Default MPLS EXP Classifier on page 47](#)
- [Applying MPLS EXP Classifiers to Routing Instances on page 50](#)

---

## Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets

On Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers and on EX Series switches, 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 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 routing devices. 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 routing device 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 {
      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 {
  ge-0/1/0 {
    unit 0 {
      family inet {
        filter input vrf-rewrite;
      }
    }
  }
}

```



**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 routing device.
- 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.

## Default IEEE 802.1ad Classifier

Table 16 on page 67 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 16: 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

**Related Documentation**

- [Configuring and Applying IEEE 802.1ad Classifiers on page 67](#)

## Configuring and Applying IEEE 802.1ad Classifiers

For Juniper Networks MX Series 3D Universal Edge Router 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 and the 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.

## Defining Custom IEEE 802.1ad Maps

You can customize the default IEEE 802.1ad map by defining values for IEEE 802.1ad code points.

```
class-of-service {
  classifiers {
    ieee-802.1ad dot1p_dei_class {
      forwarding-class best-effort {
        loss-priority low code-points [ 0000 1101 ];
      }
    }
  }
}
```

## Applying Custom IEEE 802.1ad Maps

You then apply the classifier map to the logical interface:

```
interfaces {
  ge-2/0/0 {
    unit 0 {
      classifiers {
        ieee-802.1ad dot1p_dei_class;
      }
    }
  }
}
```

## Verifying Custom IEEE 802.1ad Map Configuration

To verify your configuration, you can issue the following operational mode commands:

- **show class-of-service forwarding-table loss-priority-map**
- **show class-of-service forwarding-table loss-priority-map mapping**
- **show chassis forwarding**
- **show pfe fwdd**

## CHAPTER 4

# Assigning Service Levels to Packets Using Multifield Classifiers

- Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields on page 69
- Configuring Multifield Classifiers on page 70
- Example: Classifying Packets Based on Their Destination Address on page 73
- Example: Configuring and Verifying a Complex Multifield Filter on page 74

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

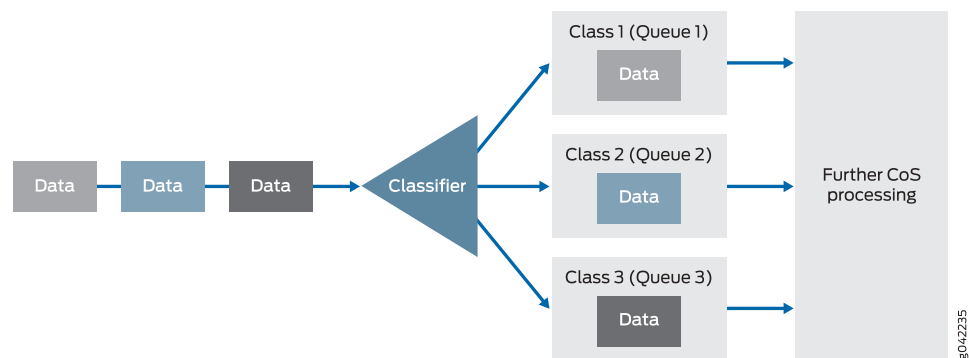
---

Behavior aggregate (BA) classification (see [“Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic” on page 26](#)), 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 7 on page 70](#) provides a high-level illustration of how a classifier works.

Figure 7: 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 26](#)
- [Configuring Multifield Classifiers on page 70](#)

## 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.



**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.  

```
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 the **loss-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.

```
[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 69](#)
- [Example: Configuring a Simple Filter on page 254](#)
- [Guidelines for Applying Firewall Filters](#)

- *Using Multifield Classifiers to Set Packet Loss Priority*

## Example: Classifying Packets Based on Their Destination Address

Configure a multifield classifier 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. Apply this filter to the inbound interface **ge-1/2/2.0**.

To verify that your configuration is attached to the correct interface, issue the **show interfaces filters** command.

```
[edit]
firewall {
  family inet {
    filter set-FC-to-platinum {
      term match-a-single-route {
        from {
          destination-address {
            10.10.10.0/24;
          }
        }
        then {
          forwarding-class platinum;
          accept;
        }
      }
      term accept-all {
        then accept;
      }
    }
  }
}
interfaces {
  ge-1/2/2 {
    unit 0 {
      family inet {
        filter {
          input set-FC-to-platinum;
        }
      }
    }
  }
}
```

## 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 74](#)
- [Verifying a Complex Multifield Filter on page 76](#)

### 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 {
          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 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*** 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
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.



## CHAPTER 5

# Controlling Access to the Network Using Traffic Policing

- [Controlling Network Access Using Traffic Policing Overview on page 79](#)
- [Effect of Two-Color Policers on Shaping Rate Changes on page 85](#)
- [Configuring Two-Color Policers and Shaping Rate Changes on page 86](#)
- [Configuring Policers Based on Logical Interface Bandwidth on page 87](#)
- [Example: Configuring a Logical Bandwidth Policer on page 89](#)
- [Overview of Tricolor Marking Architecture on page 90](#)
- [Tricolor Marking Limitations on page 91](#)
- [Configuring Tricolor Marking on page 92](#)
- [Configuring Single-Rate Tricolor Marking on page 93](#)
- [Configuring Two-Rate Tricolor Marking on page 96](#)
- [Enabling Tricolor Marking and Limitations of Three-Color Policers on page 100](#)
- [Configuring and Applying Tricolor Marking Policers on page 102](#)
- [Applying Tricolor Marking Policers to Firewall Filters on page 107](#)
- [Applying Firewall Filter Tricolor Marking Policers to Interfaces on page 108](#)

## Controlling Network Access Using Traffic Policing Overview

---

This topic covers the following information:

- [Congestion Management for IP Traffic Flows on page 79](#)
- [Traffic Limits on page 80](#)
- [Traffic Color Marking on page 81](#)
- [Forwarding Classes and PLP Levels on page 83](#)
- [Policer Application to Traffic on page 84](#)

## 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

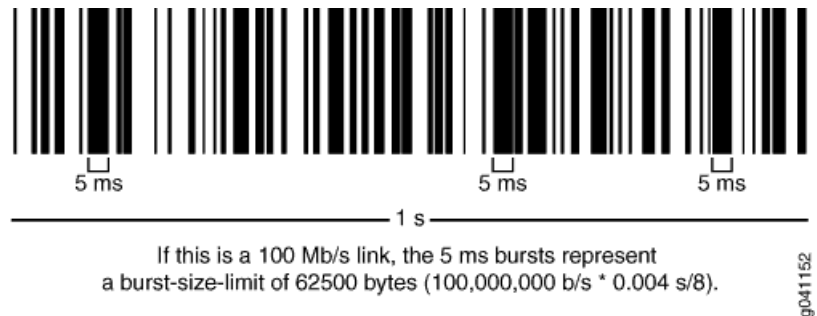
Junos OS policers use a *token bucket algorithm* to enforce a limit on an average transmit or receive rate of traffic at an interface while allowing bursts of traffic up to a maximum value based on the configured bandwidth limit and configured burst size. The token bucket algorithm offers more flexibility 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 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 8: 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), 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 17 on page 82](#) describes the policer actions.

**Table 17: 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
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

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

Three-color policers are not bound by a green-yellow-red coloring convention. Packets are marked with low, medium-high, or high PLP bit configurations based on color, so both three-color policer schemes extend the functionality of class-of-service (CoS) traffic policing by providing three levels of drop precedence (loss priority) instead of the two normally available in port-level policers. Both single-rate and two-rate three-color policer schemes can operate in two modes:

- *Color-blind*—In color-blind mode, the three-color policer assumes that all packets examined have not been previously marked or metered. In other words, the three-color policer is “blind” to any previous coloring a packet might have had.
- *Color-aware*—In color-aware mode, the three-color policer assumes that all packets examined have been previously marked or metered. In other words, the three-color policer is “aware” of the previous coloring a packet might have had. In color-aware mode, the three-color policer can increase the PLP of a packet, but never decrease it. For example, if a color-aware three-color policer meters a packet with a medium PLP marking, it can raise the PLP level to high, but cannot reduce the PLP level to low.



**NOTE:** We recommend you use the naming convention *policertypeTCM#-color type* when configuring three-color policers and *policer#* when configuring two-color policers. TCM stands for three-color marker. Because policers can be numerous and must be applied correctly to work, a simple naming convention makes it easier to apply the policers properly.

For example, the first single-rate, color-aware three-color policer configured would be named *srTCM1-ca*. The second two-rate, color-blind three-color policer configured would be named *trTCM2-cb*.

## 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 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 16](#)

## 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.

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 87](#)

---

## Configuring Two-Color Policers and Shaping Rate Changes

In the following configuration, the shaping rate has been configured for the logical interface, but a bandwidth percentage policer is also configured. Therefore policing is based on the physical interface speed of 1 Gbps.

If both a shaping rate and a bandwidth percentage policer are configured on the same logical interface, the policing is based on the physical interface speed. Here is the example configuration:

```
[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;
      }
    }
  }
}
```

```

    }
  }
}

```

## 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

```

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 172.1.1.1/30
```

- 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 172.2.1.1/30
```

#### 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;
}

[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 172.1.1.1/30;
}
```

```

    }
  }
  unit 1 {
    vlan-id 200;
    family inet {
      policer {
        input Logical_Policer;
        output Logical_Policer;
      }
      address 172.2.1.1/30;
    }
  }
}

```

5. Save the configuration.

```

[edit]
user@host# commit

```

#### Related Documentation

- [Controlling Network Access Using Traffic Policing Overview on page 79](#)
- [logical-bandwidth-policer](#)
- [shaping-rate \(Applying to an Interface\) on page 553](#)

### Example: Configuring a Logical Bandwidth Policer

This example 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).

```

[edit firewall]
policer Logical_Policer {
  logical-bandwidth-policer; # This applies the policer to logical interfaces
  if-exceeding {
    bandwidth-percent 50; # This applies 50 percent to the shaping-rate
    burst-size-limit 125k;
  }
  then discard;
}

[edit class-of-service]
interfaces {
  ge-0/2/7 {
    unit 0 {
      shaping-rate 4m # This establishes the rate to be policed on unit 0
    }
    unit 1 {
      shaping-rate 2m # This establishes the rate to be policed on unit 1
    }
  }
}

[edit 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 172.1.1.1/30;
}
}
unit 1 {
  vlan-id 200;
  family inet {
    policer {
      input Logical_Policer;
      output Logical_Policer;
    }
    address 172.2.1.1/30;
  }
}
}

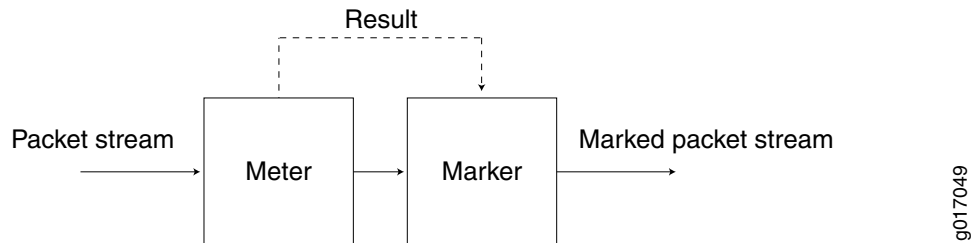
```

## 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 9 on page 90](#).

**Figure 9: 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 96](#).

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.

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 36](#) and [Using Multifield Classifiers to Set Packet Loss Priority](#).

- Related Documentation**
- [Enabling Tricolor Marking and Limitations of Three-Color Policers on page 100](#)
  - [Configuring and Applying Tricolor Marking Policers on page 102](#)

## Tricolor Marking Limitations

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

The following limitations apply to TCM:

- When you enable TCM on a 10-port Gigabit Ethernet PIC or a 10-Gigabit Ethernet PIC, for queues 6 and 7 only, the output of the **show interfaces queue *interface-name*** command does not display the number of queued bytes and packets, or the number of bytes and packets dropped due to RED. If you do not configure tricolor marking on the interface, these statistics are available for all queues.
- When you enable TCM, Transmission Control Protocol (TCP)-based configurations for drop profiles are rejected. In other words, you cannot include the **protocol** statement at the **[edit class-of-service schedulers *scheduler-name* drop-profile-map]** hierarchy level. The result is that drop profiles are applied to packets with the specified PLP and any protocol type.
- On Gigabit Ethernet IQ PICs, for IEEE 802.1 rewrite rules, only two loss priorities are supported. Exiting packets with medium-high loss priority are treated as high, and packets with medium-low loss priority are treated as low. In other words rewrite rules corresponding to high and low apply instead of those corresponding to medium-high and medium-low. For IQ PICs, you can only configure one IEEE 802.1 rewrite rule on a physical port. All logical ports (units) on that physical port should apply the same IEEE 802.1 rewrite rule.
- When some PICs with Frame Relay encapsulation mark a packet with high loss priority, the packet is treated as having medium-high loss priority on M320 Multiservice Edge Routers and T Series Core Routers with Enhanced II FPCs and T640 Core Routers with Enhanced Scaling FPC4.
- In a single firewall filter term, you cannot configure both the **loss-priority** action modifier and the **three-color-policer** action modifier. These statements are mutually exclusive.

## Configuring Tricolor Marking

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

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

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

```

        peak-burst-size bytes;
    }
}
filter filter-name {
  <family family> {
    term rule-name {
      then {
        three-color-policer (single-rate | two-rate) policer-name;
      }
    }
  }
}

```

**Related Documentation**

- [Controlling Network Access Using Traffic Policing Overview on page 79](#)

## 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.

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 93](#)
- [Configuring Color-Aware Mode for Single-Rate Tricolor Marking on page 94](#)

### 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 18 on page 93](#).

**Table 18: Color-Blind Mode TCM Color-to-PLP Mapping**

Color	PLP	Meaning
Green	<b>low</b>	Packet does not exceed the CBS.
Yellow	<b>medium-high</b>	Packet exceeds the CBS but does not exceed the EBS.
Red	<b>high</b>	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 19 on page 94](#).

Table 19: Color-Aware Mode TCM PLP Mapping

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

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

### Effect on Low PLP of Single-Rate Policer

Packets belonging to the green class have already been marked by a classifier with low PLP. The marking policer can leave the packet's PLP unchanged or increase the PLP to

medium-high or high. Therefore, these packets are metered against both the CBS and the EBS.

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

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

---

#### Effect on Medium-Low PLP of Single-Rate Policer

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

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

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

---

#### Effect on Medium-High PLP of Single-Rate Policer

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

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

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

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

### Effect on High PLP of Single-Rate Policer

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

#### Related Documentation

- [Configuring and Applying Tricolor Marking Policers on page 102](#)
- [Configuring Two-Rate Tricolor Marking on page 96](#)

## 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 96](#)
- [Configuring Color-Aware Mode for Two-Rate Tricolor Marking on page 97](#)

### 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 20 on page 96](#).

**Table 20: 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* 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 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 21 on page 97](#).

Table 21: 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.
- 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 102](#)
- [Configuring Single-Rate Tricolor Marking on page 93](#)

## Enabling Tricolor Marking and Limitations of Three-Color Policers

---

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 22 on page 101 lists the default state for TCM on Juniper Networks devices:

**Table 22: 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

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 90](#)
- [Configuring and Applying Tricolor Marking Policers on page 102](#)

---

## 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 102](#)
- [Applying Tricolor Marking Policers to Firewall Filters on page 104](#)
- [Applying Firewall Filter Tricolor Marking Policers to Interfaces on page 105](#)
- [Example: Configuring and Applying a Single-Rate Tricolor Marking Policer on page 106](#)

### 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 100):

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).

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

- Low
- Medium-high
- High

Table 23 on page 103 describes all the configurable TCM statements.

**Table 23: Tricolor Marking Policer Statements**

Statement	Meaning	Configurable Values
<b>single-rate</b>	Marking is based on the CIR, CBS, and EBS.	—
<b>two-rate</b>	Marking is based on the CIR, PIR, and rated burst sizes.	—
<b>color-aware</b>	Metering depends on the packet's preclassification. Metering can increase a packet's assigned PLP, but cannot decrease it.	—
<b>color-blind</b>	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.	—
<b>committed-information-rate</b>	Guaranteed bandwidth under normal line conditions and the average rate up to which packets are marked green.	1500 through 100,000,000,000 bps
<b>committed-burst-size</b>	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
<b>excess-burst-size</b>	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
<b>peak-information-rate</b>	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
<b>peak-burst-size</b>	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**

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;
  }
}
```

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.

- 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 srtcm1-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.

```
[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 79](#)
- [Overview of Tricolor Marking Architecture on page 90](#)
- [Configuring and Applying Tricolor Marking Policers on page 102](#)

## Applying Tricolor Marking Policers to Firewall Filters

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

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

You can include this statement at the following hierarchy levels:

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

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

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

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

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

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

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

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

#### Related Documentation

- *Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices*

---

### Applying Firewall Filter Tricolor Marking Policers to Interfaces

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

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

You can include these statements at the following hierarchy levels:

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

The filter name that you reference should have an attached tricolor marking policer, as shown in “[Applying Tricolor Marking Policers to Firewall Filters](#)” on page 107.

### 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;
            }
        }
    }
}
}
```



## CHAPTER 6

# Defining Forwarding Behavior Based on Forwarding Classes

- [Forwarding Classes Overview on page 111](#)
- [Default Forwarding Classes on page 114](#)
- [Configuring Forwarding Classes on page 116](#)
- [Configuring Up to 16 Forwarding Classes on page 117](#)
- [Classifying Packets by Egress Interface on page 124](#)
- [Forwarding Policy Options Overview on page 126](#)
- [Configuring CoS-Based Forwarding on page 126](#)
- [Example: Configuring CoS-Based Forwarding on page 129](#)
- [Example: Configuring CoS-Based Forwarding for Different Traffic Types on page 132](#)
- [Example: Configuring CoS-Based Forwarding for IPv6 on page 132](#)
- [Overriding the Input Classification on page 133](#)
- [Applying Forwarding Classes to Interfaces on page 134](#)
- [Default Routing Engine Protocol Queue Assignments on page 135](#)
- [Changing the Default Queuing and Marking of Host Outbound Traffic on page 137](#)
- [Assigning Forwarding Class and DSCP Value for Routing Engine-Generated Traffic on page 138](#)
- [Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 139](#)

## Forwarding Classes Overview

---

This topic covers the following information:

- [Output Queue Assignments Based on Forwarding Class on page 112](#)
- [Devices That Support Up to Four Forwarding Classes on page 112](#)
- [Devices That Support Up to 16 Forwarding Classes on page 113](#)
- [Default and Configurable Packet Loss Priority Values on page 113](#)
- [Configuration Statements Used to Configure and Apply Forwarding Classes on page 113](#)

## 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.



**NOTE:** PTX1000 supports the following default forwarding classes:

- Best effort (BE)
  - Fibre Channel over Ethernet (FCoE)
  - No Loss
  - Network control (NC)
- 

## Devices That Support Up to Four Forwarding Classes

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

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



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

---

## Devices That Support Up to 16 Forwarding Classes

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



**NOTE:** PTX1000 routers support only 8 unicast forwarding classes.

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

## Default and Configurable Packet Loss Priority Values

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

- M320 routers and T Series routers with Enhanced III Flexible PIC Concentrators (FPCs)
- T640 routers with Enhanced Scaling FPC4s
- PTX Series Packet Transport Routers



**NOTE:** medium-low priority is not supported on PTX1000 routers.

## Configuration Statements Used to Configure and Apply Forwarding Classes

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

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

```

}
forwarding-class-map forwarding-class-map-name {
    class class-name queue-num queue-number [ restricted-queue queue-number ];
}
interfaces {
    interface-name {
        unit logical-unit-number {
            forwarding-class class-name;
            forwarding-class-map forwarding-class-map-name;
        }
    }
}
restricted-queues {
    forwarding-class class-name queue queue-number;
}

```

#### Related Documentation

- *Default Forwarding Classes*
- [Configuring Forwarding Classes on page 116](#)
- [Applying Forwarding Classes to Interfaces on page 134](#)
- *Configuring Up to 16 Forwarding Classes*
- *Policer Overview*

## Default Forwarding Classes

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

These default mappings apply to all routers. The four forwarding classes defined by default are shown in [Table 24 on page 114](#).

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

Some routers support eight queues. Queues 4 through 7 have no default mappings to forwarding classes. To use queues 4 through 7, you must create custom forwarding class names and map them to the queues. For more information, see the Juniper Networks J Series Services Router documentation.

**Table 24: Default Forwarding Classes**

Queue	Forwarding Class Name	Comments
Queue 0	<b>best-effort (be)</b>	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.

Table 24: Default Forwarding Classes (*continued*)

Queue	Forwarding Class Name	Comments
Queue 1	<b>expedited-forwarding (ef)</b>	<p>The software delivers assured bandwidth, low loss, low delay, and low delay variation (jitter) end-to-end for packets in this service class.</p> <p>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.</p>
Queue 2	<b>assured-forwarding (af)</b>	<p>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.</p> <p>The software accepts excess traffic, but applies a RED drop profile to determine if the excess packets are dropped and not forwarded.</p> <p>Depending on router type, up to four drop probabilities (low, medium-low, medium-high, and high) are defined for this service class.</p>
Queue 3	<b>network-control (nc)</b>	<p>The software delivers packets in this service class with a low priority. (These packets are not delay sensitive.)</p> <p>Typically, these packets represent routing protocol hello or keepalive messages. Because loss of these packets jeopardizes proper network operation, delay is preferable to discard.</p>

The following rules govern queue assignment:

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

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

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

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

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

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

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

- 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.

---

## Configuring Forwarding Classes

---

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

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



**NOTE:** Committing changes to schedulers and queues interrupts traffic on affected ports while queue resources are reconfigured.

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.

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.

#### Related Documentation

- [Forwarding Classes Overview on page 111](#)
- [Default Forwarding Classes](#)
- [Changing the Default Queuing and Marking of Host Outbound Traffic on page 137](#)

## Configuring Up to 16 Forwarding Classes

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



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

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

Mapping multiple forwarding classes to single queues is useful. Suppose, for example, that forwarding classes are set based on multifield packet classification, and the multifield classifiers are different for core-facing interfaces and customer-facing interfaces. Suppose you need four queues for a core-facing interface and five queues for a customer-facing interface, where **fc0** through **fc4** correspond to the classifiers for the customer-facing interface, and **fc5** through **fc8** correspond to classifiers for the core-facing interface, as shown in [Figure 10 on page 118](#).

**Figure 10: Customer-Facing and Core-Facing Forwarding Classes**



g016702

In this example, there are nine classifiers and, therefore, nine forwarding classes. The forwarding class-to-queue mapping is shown in [Table 25 on page 118](#).

**Table 25: Sample Forwarding Class-to-Queue Mapping**

Forwarding Class Names	Queue Number
fc0	0
fc5	
fc1	1
fc6	
fc2	2
fc7	
fc3	3
fc8	
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 to 16 different forwarding-class names. The corresponding output queue number can be from 0 through 7. Therefore, you can map multiple forwarding

classes to a single queue. If you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler (at the **[edit class-of-service scheduler-maps map-name forwarding-class class-name scheduler scheduler-name]** hierarchy level).

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

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

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

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

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

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

You can determine the ID number assigned to a forwarding class by issuing the **show class-of-service forwarding-class** command. You can determine whether the classification is fixed by issuing the **show class-of-service forwarding-table classifier mapping** command. In the command output, if the **Table Type** field appears as **Fixed**, the classification is fixed. For more information about fixed classification, see [“Applying Forwarding Classes to Interfaces” on page 134](#).

This section discusses the following topics:

- [Enabling Eight Queues on Interfaces on page 120](#)
- [Multiple Forwarding Classes and Default Forwarding Classes on page 121](#)
- [PICs Restricted to Four Queues on page 121](#)
- [Examples: Configuring Up to 16 Forwarding Classes on page 122](#)

## Enabling Eight Queues on Interfaces

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

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

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

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

The numerical value can be 4 or 8.

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



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

---

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

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

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

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

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

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

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

## Multiple Forwarding Classes and Default Forwarding Classes

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

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

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

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

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

For more information about forwarding class defaults, see *Default Forwarding Classes*.

## PICs Restricted to Four Queues

Some Juniper Networks T Series Core Router PICs support up to 16 forwarding classes and are restricted to 4 queues. Contact Juniper Networks customer support for a current list of T Series router PICs that are restricted to four queues. To determine how many

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

```
user@host> show interfaces ge-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:

$$Q_r = Q_d \bmod R_{max}$$

**Q<sub>r</sub>** is the queue number assigned if the PIC is restricted to four queues.

**Q<sub>d</sub>** is the queue number that would have been mapped if this PIC were not restricted.

**R<sub>max</sub>** 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 **Q<sub>r</sub> = 6 mod 4 = 2**.

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

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

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

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

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

## Examples: Configuring Up to 16 Forwarding Classes

Configure 16 forwarding classes:

### Configuring 16 Forwarding Classes

```
[edit class-of-service]
forwarding-classes {
  class fc0 queue-num 0;
  class fc1 queue-num 0;
  class fc2 queue-num 1;
```

```

class fc3 queue-num 1;
class fc4 queue-num 2;
class fc5 queue-num 2;
class fc6 queue-num 3;
class fc7 queue-num 3;
class fc8 queue-num 4;
class fc9 queue-num 4;
class fc10 queue-num 5;
class fc11 queue-num 5;
class fc12 queue-num 6;
class fc13 queue-num 6;
class fc14 queue-num 7;
class fc15 queue-num 7;
}

```

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

**Restricted Queues:  
Mapping Two  
Forwarding Classes to  
Each Queue**

```

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

```

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

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

```

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

```

```
forwarding-class ef1 queue 1;
forwarding-class ef2 queue 1;
forwarding-class af queue 2;
forwarding-class af1 queue 2;
forwarding-class nc queue 3;
forwarding-class nc1 queue 3;
}
```

## Classifying Packets by Egress Interface

---

For Juniper Networks EX Series switches, M320 Multiservice Edge Routers, and T Series Core Routers with the Intelligent Queuing (IQ), IQ2, Enhanced IQ (IQE), or Multiservices link services intelligent queuing (LSQ) interfaces, you can classify unicast and multicast packets based on the egress interface. For unicast traffic, you can also use a multifeild filter, but only egress interface classification applies to multicast traffic as well as unicast traffic. If you configure egress classification of an interface, you cannot perform Differentiated Services code point (DSCP) rewrites on the interface. By default, the system will not perform any classification based on the egress interface.

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

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

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



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

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

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

All parameters relating to the queues and forwarding class must be configured as well. For more information about configuring forwarding classes and queues, see [“Configuring Forwarding Classes” on page 116](#).

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-classes-interface-specific FCMAP1 {
  class FC1 queue-num 6 restricted-queue 3;
  class FC2 queue-num 5 restricted-queue 2;
  class FC3 queue-num 3;
  class FC4 queue-num 0;
  class FC3 queue-num 0;
  class FC4 queue-num 1;
}

[edit class-of-service]
interfaces {
  ge-6/0/0 unit 0 {
    output-forwarding-class-map FCMAP1;
  }
}
```

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

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

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

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

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

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

Physical interface: ge-6/0/0, Index: 128

Queues supported: 8, Queues in use: 8

Scheduler map: <default>, Index: 2

Input scheduler map: <default>, Index: 3

Chassis scheduler map: <default-chassis>, Index: 4

Logical interface: ge-6/0/0.0, Index: 67

Object	Name	Type	Index
Scheduler-map	sch-map1	Output	6998
Scheduler-map	sch-map1	Input	6998
Classifier	dot1p	ieee8021p	4906
forwarding-class-map	FCMAP1	Output	1221

Logical interface: ge-6/0/0.1, Index 68

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

---

## Forwarding Policy Options Overview

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

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

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

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

---

## 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 for Routing Devices*.



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

You can configure CBF on a routing device with eight or 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 eight. Similarly, with CBF configured, you cannot configure more than eight 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;
  }
}
```



**NOTE:** CBF is not supported on PTX1000 routers.

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

For Layer 3 VPNs, when you use class-based forwarding for the routes received from the far-end provider edge (PE) router within a VRF instance, the software can match the routes based on the attributes that come with the received route only. In other words,

the matching can be based on the route within RIB-in. In this case, the **route-filter** statement you include at the **[edit policy-options policy-statement my-cos-forwarding from]** hierarchy level has no effect because the policy checks the **bgp.l3vpn.0** table, not the **vrf.inet.0** table.

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 QN are a set of forwarding classes that map to the specific next hop. That is, when a packet is switched with Q1 through QN, 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

**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, one is chosen randomly.

#### Related Documentation

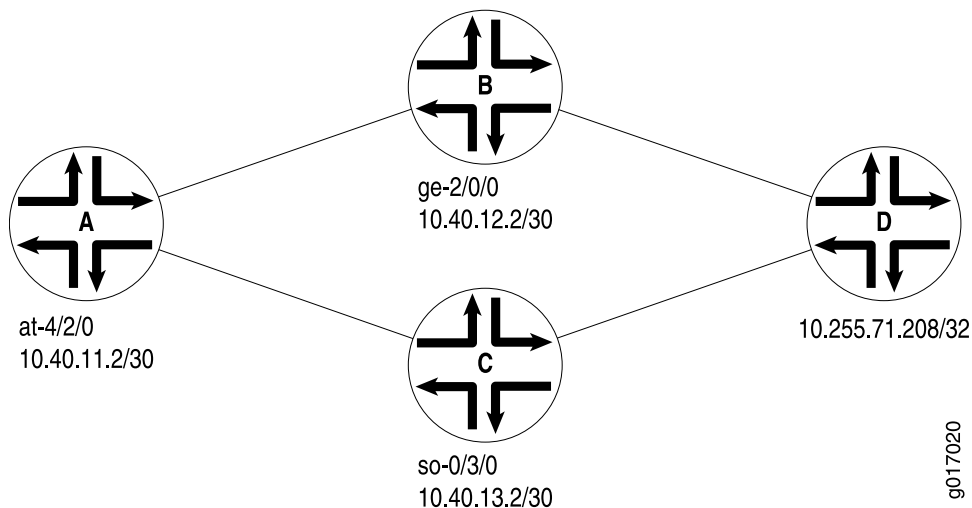
- *Load Balancing VPLS Non-Unicast Traffic Across Member Links of an Aggregate Interface*

## 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 11 on page 130](#).

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

Figure 11: Sample CoS-Based Forwarding



When you configure CBF with OSPF as the IGP, you must specify the next hop as an interface name, not as an IP address. The next hops in this example are specified as **ge-2/0/0.0** and **ge-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 ge-0/3/0.0;
    }
    forwarding-class af {
      next-hop ge-2/0/0.0;
    }
    forwarding-class nc {
      next-hop ge-0/3/0.0;
    }
  }
}
classifiers {
  inet-precedence inet {
    forwarding-class be {
      loss-priority low code-points [ 000 100 ];
    }
    forwarding-class ef {
      loss-priority low code-points [ 001 101 ];
    }
    forwarding-class af {
      loss-priority low code-points [ 010 110 ];
    }
    forwarding-class nc {
      loss-priority low code-points [ 011 111 ];
    }
  }
}

```

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

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

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

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

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

---

### Example: Configuring CoS-Based Forwarding for Different Traffic Types

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

Only the forwarding policy is shown in this example:

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

---

### Example: Configuring CoS-Based Forwarding for IPv6

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

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

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

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

1. Define the CBF next-hop map:

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

2. Define the CBF forwarding policy:

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

3. Export the CBF forwarding policy:

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

---

## Overriding the Input Classification

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

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

To override the input packet classification, do the following:

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

---

## Applying Forwarding Classes to Interfaces

---

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



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

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

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

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

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

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

**Related Documentation**

- [forwarding-class on page 354](#)

## Default Routing Engine Protocol Queue Assignments

[Table 26 on page 135](#) lists (in alphabetical order) how Routing Engine-sourced traffic is mapped to output queues. The following caveats apply to [Table 26 on page 135](#):

- For all packets sent to queue 3 over a VLAN-tagged interface, the software sets the 802.1p bit to 110.
- 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 26: Routing Engine Protocol Queue Assignments**

Routing Engine Protocol	Queue Assignment
Adaptive Services PIC	TCP tickle (keepalive packets for idle session generated with stateful firewall to probe idle TCP sessions) are sent from queue 0.
Bidirectional Forwarding Detection (BFD) Protocol	Queue 3
Border Gateway Protocol (BGP)	Queue 0
BGP TCP Retransmission	Queue 3
Cisco High-Level Data Link Control (HDLC)	Queue 3
Distance Vector Multicast Routing Protocol (DVMRP)	Queue 3
Frame Relay Local Management Interface (LMI)	Queue 3
Frame Relay Asynchronization permanent virtual circuit (PVC)/data link connection identifier (DLCI) status messages	Queue 3

**Table 26: Routing Engine Protocol Queue Assignments (*continued*)**

Routing Engine Protocol	Queue Assignment
FTP	Queue 0
Intermediate System-to-Intermediate System (IS-IS) Open Systems Interconnection (OSI)	Queue 3
Internet Group Management Protocol (IGMP) query	Queue 3
IGMP Report	Queue 0
IP version 6 (IPv6) Neighbor Solicitation	Queue 3
IPv6 Neighbor Advertisement	Queue 3
IPv6 Router Advertisement	Queue 0
Label Distribution Protocol (LDP) User Datagram Protocol (UDP) hello	Queue 3
LDP keepalive and Session data	Queue 0
LDP TCP Retransmission	Queue 3
Link Aggregation Control Protocol (LACP)	Queue 3
Link Services (LS) PIC	If link fragmentation and interleaving (LFI) is enabled, all routing protocol packets larger than 128 bytes are transmitted from queue 0. This ensures that VoIP traffic is not affected. Fragmentation is supported on queue 0 only.
Multicast listener discovery (MLD)	Queue 0
Multicast Source Discovery Protocol (MSDP)	Queue 0
MSDP TCP Retransmission	Queue 3
Multilink Frame Relay Link Integrity Protocol (LIP)	Queue 3
Open Shortest Path First (OSPF) protocol data unit (PDU)	Queue 3
Protocol Independent Multicast (PIM)	Queue 3
Real-time performance monitoring (RPM) probe packets	Queue 3
Resource Reservation Protocol (RSVP)	Queue 3

Table 26: Routing Engine Protocol Queue Assignments (*continued*)

Routing Engine Protocol	Queue Assignment
Routing Information Protocol (RIP)	Queue 3
Simple Network Management Protocol (SNMP)	Queue 0
SSH	Queue 0
Telnet	Queue 0
Virtual Router Redundancy Protocol (VRRP)	Queue 3
xnm-clear-text	Queue 0
xnm-ssl	Queue 0

## 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:

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

- Related Documentation**
- [Forwarding Classes Overview on page 111](#)
  - [Default Queue Assignments for Routing Engine Sourced Traffic](#)
  - [Default DSCP and DSCP IPv6 Classifiers on page 42](#)

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

---

You can set the forwarding class and differentiated service code point (DSCP) value for traffic originating in the Routing Engine. To configure forwarding class and DSCP values that apply to Routing Engine-generated traffic only, apply an output filter to the loopback (**lo.0**) interface and set the appropriate forwarding class and DSCP bit configuration for various protocols. For example, you can set the DSCP value on OSPF packets that originate in the Routing Engine to **10** and assign them to the AF (assured forwarding) forwarding class while the DSCP value on ping packets are set to **0** and use forwarding class BE (best effort).

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

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

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

[edit firewall filter family inet]
filter loopback-filter {
  term t1 {
    from {
      protocol icmp; # For pings
    }
    then {
      forwarding-class af17;
      dscp 38;
    }
  }
  term t2 {
    from {
      protocol ospf; # For OSPF
    }
    then {
```

```

        forwarding-class af11;
        dscp 12;
    }
}
term t3 {
    from {
        protocol tcp; # For BGP
    }
    then {
        forwarding-class af16;
        dscp 10;
    }
}
term t4 {
    then accept; # Do not forget!
}
}

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

```



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

## 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 26 and [“Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields”](#) on page 69.

However, on Juniper Networks M320 Multiservice Edge Routers, MX Series 3D Universal Edge Routers, 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 III Flexible PIC Concentrators (FPCs) and tricolor marking enabled, you can set the PLP with a BA or multifield classifier, as

described in [“Using BA Classifiers to Set PLP” on page 35](#) and *Using Multifield Classifiers to Set Packet Loss Priority*.

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.

### Example: Overriding the Default PLP on M320 Routers

The following example shows a two-step procedure to override the default PLP settings on M320 routers:

1. The following 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.

```
class-of-service {
  classifiers {
    dscp ba-classifier {
      forwarding-class expedited-forwarding {
        loss-priority high code-points 110;
      }
    }
  }
}
```

2. For M320 routers, this multifield classifier sets the PLP.

```
firewall {
  filter ef-filter {
    term ef-multifield {
      from {
        precedence 6;
      }
      then {
        loss-priority high;
        forwarding-class expedited-forwarding;
      }
    }
  }
}
```

### 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.

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

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

#### Related Documentation

- [Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 205](#)
- [Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199](#)
- [Configuring Schedulers on page 146](#)



## CHAPTER 7

# Defining Output Queue Properties Using Schedulers

- [Schedulers Overview on page 143](#)
- [Default Schedulers Overview on page 145](#)
- [Configuring Schedulers on page 146](#)
- [Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size on page 146](#)
- [Configuring Scheduler Maps on page 157](#)
- [Applying Scheduler Maps to Physical Interfaces on page 158](#)
- [Priority Scheduling Overview on page 158](#)
- [Applying Scheduler Maps Overview on page 159](#)
- [Applying a Shaping Rate to Physical Interfaces Overview on page 160](#)
- [Forwarding Classes and Fabric Priority Queues on page 161](#)

## Schedulers Overview

---

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.

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;
        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)

```

## 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 and assured-forwarding classes have no schedulers because, by default, no resources are assigned to queue 1 and queue 2. 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 174](#).

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;
  }
}
```

## 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;
    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 146](#)
- [Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199](#)
- [Configuring Scheduler Transmission Rate on page 172](#)
- [Configuring Schedulers for Priority Scheduling on page 195](#)

## 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, 5, 0, 0, 0, and 0 percent of the total available bandwidth.

The default buffer size percentages for queues 0 through 7 are 95, 0, 0, 5, 0, 0, 0, and 0 percent of the total available buffer. The total available buffer per queue differs by PIC type, as shown in [Table 27 on page 147](#).

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 | 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 27 on page 147](#).
- 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.
- 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 27 on page 147](#). The maximums apply to the logical interface, not each queue.



**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.

For information about configuring large buffer sizes on IQ PICs, see [“Configuring Large Delay Buffers for Slower Interfaces” on page 148](#).

**Table 27: 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

Table 27: Buffer Size Temporal Value Ranges by Routing Device Type (*continued*)

Routing Devices	Temporal Value Ranges
<b>With Large Buffer Sizes Enabled</b>	
IQ PICs on all routers	1 through 500,000 microseconds
<b>Gigabit Ethernet IQ VLANs</b>	
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 148](#)
- [Enabling and Disabling the Memory Allocation Dynamic per Queue on page 156](#)

### Configuring Large Delay Buffers for Slower Interfaces

By default, T1, E1, and NxDS0 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 28 on page 149](#) shows some recommended buffer sizes needed to absorb typical burst sizes from various upstream interface types.

Table 28: 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 NxDS0 interfaces and DLCIs, you can configure a buffer size larger than the default maximum. To enable larger buffer sizes to be configured, include the **q-pic-large-buffer (large-scale | small-scale)** 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 large-scale;
```

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 29 on page 149](#).

Table 29: Maximum Delay Buffer with q-pic-large-buffer Enabled by Interface

Platform, PIC, or Interface Type	Maximum Buffer Size
<b>With Large Buffer Sizes Not Enabled</b>	
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
<b>With Large Buffer Sizes Enabled</b>	
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

**Table 29: Maximum Delay Buffer with q-pic-large-buffer Enabled by Interface** (*continued*)

Platform, PIC, or Interface Type	Maximum Buffer Size
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
NxDSO IQ Interfaces—Maximum sizes vary by channel size:	
1xDSO through 3xDSO	4,000,000 microseconds
4xDSO through 7xDSO	2,000,000 microseconds
8xDSO through 15xDSO	1,000,000 microseconds
16xDSO through 32xDSO	500,000 microseconds
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 181](#).

For NxDSO interfaces, the larger buffer sizes can be up to 4,000,000 microseconds, depending on the number of DSO channels in the NxDSO interface. For slower NxDSO interfaces with fewer channels, the delay buffer can be relatively larger than for faster NxDSO interfaces with more channels. This is shown in [Table 31 on page 152](#). To calculate specific buffer sizes for various NxDSO interfaces, see [“Maximum Delay Buffer for NxDSO Interfaces” on page 151](#).

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 30 on page 151](#).

Table 30: Delay-Buffer Calculations

Delay Buffer Configuration	Formula	Example
Percentage	$\text{available interface bandwidth} * \text{configured percentage buffer-size} * \text{maximum buffer} = \text{queue buffer}$	<p>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:</p> <pre> sched-expedited {   transmit-rate percent 30;   buffer-size percent 30; } </pre> <p> <math>1.5 \text{ Mbps} * 0.3 * 500,000 \text{ microseconds} = 225,000 \text{ bits}</math>  <math>= 28,125 \text{ bytes}</math> </p>
Temporal	$\text{available interface bandwidth} * \text{configured percentage transmit-rate} * \text{configured temporal buffer-size} = \text{queue buffer}$	<p>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:</p> <pre> sched-best {   transmit-rate percent 20;   buffer-size temporal 500000; } </pre> <p> <math>1.5 \text{ Mbps} * 0.2 * 500,000 \text{ microseconds} = 150,000 \text{ bits}</math>  <math>= 18,750 \text{ bytes}</math> </p>
Percentage, with buffer size larger than transmit rate		<p>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.</p> <pre> sched-extra-buffer {   transmit-rate percent 10;   buffer-size percent 20; } </pre>
FRF.16 LSQ bundles	<p>For total bundle bandwidth &lt; T1 bandwidth, the delay-buffer rate is 1 second.</p> <p>For total bundle bandwidth &gt;= T1 bandwidth, the delay-buffer rate is 200 milliseconds (ms).</p>	

For more information, see the following sections:

- [Maximum Delay Buffer for NxDS0 Interfaces on page 151](#)
- [Example: Configuring Large Delay Buffers for Slower Interfaces on page 153](#)

### Maximum Delay Buffer for NxDS0 Interfaces

Because NxDS0 interfaces carry less bandwidth than a T1 or E1 interface, the buffer size on an NxDS0 interface can be relatively larger, depending on the number of DS0 channels combined. The maximum delay buffer size is calculated with the following formula:

$$\text{Interface Speed} * \text{Maximum Delay Buffer Time} = \text{Delay Buffer Size}$$

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):

$$64 \text{ Kbps} * 4,000,000 \text{ microseconds} = 32 \text{ KB}$$

Table 31 on page 152 shows the delay-buffer calculations for 1xDS0 through 32xDS0 interfaces.

**Table 31: NxDS0 Transmission Rates and Delay Buffers**

Interface Speed	Delay Buffer Size
<b>1xDS0 Through 4xDS0: Maximum Delay Buffer Time Is 4,000,000 Microseconds</b>	
1xDS0: 64 Kbps	32 KB
2xDS0: 128 Kbps	64 KB
3xDS0: 192 Kbps	96 KB
<b>4xDS0 Through 7xDS0: Maximum Delay Buffer Time Is 2,000,000 Microseconds</b>	
4xDS0: 256 Kbps	64 KB
5xDS0: 320 Kbps	80 KB
6xDS0: 384 Kbps	96 KB
7xDS0: 448 Kbps	112 KB
<b>8xDS0 Through 15xDS0: Maximum Delay Buffer Time Is 1,000,000 Microseconds</b>	
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: 896 Kbps	112 KB
15xDS0: 960 Kbps	120 KB
<b>16xDS0 Through 32xDS0: Maximum Delay Buffer Time Is 500,000 Microseconds</b>	
16xDS0: 1024 Kbps	64 KB

**Table 31: NxDSO Transmission Rates and Delay Buffers (*continued*)**

Interface Speed	Delay Buffer Size
17xDSO: 1088 Kbps	68 KB
18xDSO: 1152 Kbps	72 KB
19xDSO: 1216 Kbps	76 KB
20xDSO: 1280 Kbps	80 KB
21xDSO: 1344 Kbps	84 KB
22xDSO: 1408 Kbps	88 KB
23xDSO: 1472 Kbps	92 KB
24xDSO: 1536 Kbps	96 KB
25xDSO: 1600 Kbps	100 KB
26xDSO: 1664 Kbps	104 KB
27xDSO: 1728 Kbps	108 KB
28xDSO: 1792 Kbps	112 KB
29xDSO: 1856 Kbps	116 KB
30xDSO: 1920 Kbps	120 KB
31xDSO: 1984 Kbps	124 KB
32xDSO: 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 30 on page 151](#).

```

chassis {
  fpc 0 {
    pic 0 {
      q-pic-large-buffer; # Enabling large delay buffer
      max-queues-per-interface 8; # Eight queues (M320, T Series, and TX Matrix routers)
    }
  }
}

```

### 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.

The 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:

$$\text{available interface bandwidth} * \text{configured percentage buffer-size} * \text{maximum buffer} = \text{queue buffer}$$

$$1.5 \text{ Mbps} * 0.2 * 500,000 \text{ microseconds} = 150,000 \text{ bits} = 18,750 \text{ bytes}$$

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:

$$\text{available interface bandwidth} * \text{configured percentage transmit-rate} * \text{configured temporal buffer-size} = \text{queue buffer}$$

$$1.5 \text{ Mbps} * 0.2 * 300,000 \text{ microseconds} = 90,000 \text{ bits} = 11,250 \text{ bytes}$$

```
[edit]
class-of-service {
  schedulers {
    sched-best {
      transmit-rate percent 10;
      buffer-size percent 20;
    }
    sched-exped {
      transmit-rate percent 20;
      buffer-size temporal 300000;
    }
  }
}
```

### 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. 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. For more information, see [Table 31 on page 152](#).

```
class-of-service {
  interfaces {
    t1-0/0/0:1 {
      shaping-rate 200k;
    }
  }
}
```

```

    }
}

```

### 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 {
        address 1.1.1.1/24;
      }
      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 Services 3D Universal Edge Routers, and T Series Core Routers and EX Series 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 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, include the **buffer-size percent** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level:

```
[edit class-of-service schedulers scheduler-name]
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:

```
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;
    }
  }
  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 350](#)
- [schedulers \(CoS\) on page 361](#)
- *q-pic-large-buffer*
- [schedulers \(CoS\) on page 361](#)

---

## 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;
  }
}
```

## Applying Scheduler Maps to Physical Interfaces

---

After you have defined a scheduler map, as described in [“Configuring Scheduler Maps” on page 157](#), you can apply it to an output interface. Include the **scheduler-map** statement at the **[edit class-of-service interfaces *interface-name*]** hierarchy level:

```
[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 181](#).



**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.

## 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 the queues, thus ensuring that queues containing important traffic are provided better access to the outgoing interface. This is accomplished through a procedure in which the software examines the priority of the queue. In addition, the software determines if the individual queue is within its defined bandwidth profile. The bandwidth profile is discussed in [“Configuring Scheduler Transmission Rate” on page 172](#). This binary decision, which is reevaluated on a regular time cycle, compares the amount of data transmitted by the queue against the amount of bandwidth allocated to it by the scheduler. 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.

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.

The software 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.

---

## 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 NxDS0 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 158](#)
- [Applying Scheduler Maps and Shaping Rate to Physical Interfaces on IQ PICs](#)
- [Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 181](#)
- [Oversubscribing Interface Bandwidth on page 163](#)
- [Providing a Guaranteed Minimum Rate on page 175](#)
- [Applying Scheduler Maps to Packet Forwarding Component Queues](#)
- [Forwarding Classes and Fabric Priority Queues on page 161](#)
- [Associating Schedulers with Fabric Priorities on page 197](#)

---

## 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 240](#)

## Forwarding Classes and Fabric Priority Queues

---

This topic covers the following information:

- [Default Fabric Priority Queuing on page 161](#)
- [Overriding Default Fabric Priority Queuing on page 161](#)

### Default Fabric Priority Queuing

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

### Overriding Default Fabric Priority Queuing

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

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

#### Related Documentation

- [Associating Schedulers with Fabric Priorities on page 197](#)



## CHAPTER 8

# Controlling Bandwidth Using Scheduler Rates

- [Oversubscribing Interface Bandwidth on page 163](#)
- [Configuring Scheduler Transmission Rate on page 172](#)
- [Providing a Guaranteed Minimum Rate on page 175](#)
- [Bandwidth Sharing on Nonqueueing Packet Forwarding Engines Overview on page 179](#)
- [Configuring Rate Limits on Nonqueueing Packet Forwarding Engines on page 179](#)
- [Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs on page 181](#)
- [Applying Scheduler Maps to Packet Forwarding Component Queues on page 188](#)

### 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 AS 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 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 181.

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.



**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 AS 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 175](#).

For more information about Gigabit Ethernet IQ2 PICs, see *CoS on Enhanced IQ2 PICs Overview*.

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 [“Configuring Large Delay Buffers for Slower Interfaces” on page 148](#) and [“Maximum Delay Buffer for NxDS0 Interfaces” on page 151](#). For an example showing how the delay-buffer rates are applied, see [“Examples: Oversubscribing Interface Bandwidth” on page 169](#).

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:

$$(\text{remaining delay-buffer rate} * \text{shaping rate}) / (\text{sum of shaping rates})$$

where the remaining delay-buffer rate is equal to:

$$(\text{interface speed}) - (\text{sum of configured delay-buffer rates})$$

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 146](#) and [“Configuring Scheduler Maps” on page 157](#).

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 [“Configuring Large Delay Buffers for Slower Interfaces” on page 148](#).

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 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 32 on page 167 shows how the bandwidth and delay buffer are allocated in various configurations.

**Table 32: 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 <code>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i>]</code> hierarchy level.	<p>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 <code>q-pic-large-buffer</code> statement. For more information, see <a href="#">“Configuring Large Delay Buffers for Slower Interfaces”</a> on page 148.</p> <p>Unshaped logical interfaces receive the remaining bandwidth and a delay buffer in proportion to the remaining bandwidth.</p>
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.
You oversubscribe the interface. You configure a shaping rate. You do not configure a guaranteed rate. You do not configure a delay-buffer rate.	<p>Logical interface receives a delay buffer based on the scaled shaping rate:</p> $\text{scaled shaping rate} = (\text{shaping-rate} * [\text{physical interface bandwidth}]) / \text{SUM}(\text{shaping-rates of all logical interfaces on the physical interface})$ <p>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.</p>

**Table 32: Bandwidth and Delay Buffer Allocations by Configuration Scenario**  
*Scenario (continued)*

Configuration Scenario	Delay Buffer Allocation
You oversubscribe the interface. You configure a shaping rate. You configure a delay-buffer rate.	<p>Logical interface receives a delay buffer based on the delay-buffer rate. For example, on IQ and IQ2 interfaces:</p> <p>delay-buffer-rate &lt;= 10 Mbps: 400-millisecond (ms) delay buffer  delay-buffer-rate &lt;= 20 Mbps: 300-ms delay buffer  delay-buffer-rate &lt;= 30 Mbps: 200-ms delay buffer  delay-buffer-rate &lt;= 40 Mbps: 150-ms delay buffer  delay-buffer-rate &gt; 40 Mbps: 100-ms delay buffer</p> <p>On LSQ DLCIs, if <b>total bundle bandwidth &lt; T1 bandwidth</b>:</p> <p>delay-buffer-rate = 1 second</p> <p>On LSQ DLCIs, if <b>total bundle bandwidth &gt;= T1 bandwidth</b>:</p> <p>delay-buffer-rate = 200 ms</p> <p>The multiplicative factor depends on whether you include the <b>q-pic-large-buffer</b> statement. For more information, see <a href="#">“Configuring Large Delay Buffers for Slower Interfaces”</a> on page 148.</p> <p>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.</p>
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.	<p>Logical interface receives a delay buffer based on the guaranteed rate.</p> <p>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).</p>



**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 the 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 [“Maximum Delay Buffer for NxDSO Interfaces” on page 151](#).

```
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;
      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;  
    }  
  }  
}
```

**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 18.18.18.2/24;  
      }  
    }  
  }  
}
```

```

    }
    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;
            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:

- **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 3D Universal Edge Routers.
- 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.

- **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 146](#).

For more information about scheduler transmission rate, see the following sections:

- [Example: Configuring Scheduler Transmission Rate on page 173](#)
- [Allocation of Leftover Bandwidth on page 174](#)

## 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.

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 195](#)
- [Schedulers Overview on page 143](#)
- [Configuring a Scheduler](#)
- [excess-rate on page 353](#)

- [schedulers on page 361](#)

## 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 AS 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.

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 AS 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*.

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 [“Configuring Large Delay Buffers for Slower Interfaces” on page 148](#) and [“Maximum Delay Buffer for NxDSO Interfaces” on page 151](#). For an example showing how the delay-buffer rates are applied, see [“Example: Providing a Guaranteed Minimum Rate” on page 178](#).

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 146](#) and [“Configuring Scheduler Maps” on page 157](#).

- 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;
```

If you do not include this statement, the delay-buffer size is more restricted. For more information, see [“Configuring Large Delay Buffers for Slower Interfaces” on page 148](#).

- 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.

- 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 33 on page 177](#) shows how the bandwidth and delay buffer are allocated in various configurations.

**Table 33: 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 <b>q-pic-large-buffer</b> statement. For more information, see <a href="#">“Configuring Large Delay Buffers for Slower Interfaces” on page 148</a> .
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 <b>q-pic-large-buffer</b> statement. For more information, see <a href="#">“Configuring Large Delay Buffers for Slower Interfaces” on page 148</a> .

## 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**

### 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:

$\text{delay-buffer-rate} < [8 \times 64 \text{ Kbps}])$ : 2 seconds of delay-buffer-rate

For more information about this calculation, see [“Maximum Delay Buffer for NxDS0 Interfaces” on page 151](#).

```
chassis {
  fpc 3 {
    pic 0 {
      q-pic-large-buffer;
    }
  }
}
interfaces {
  tl-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 {
    tl-3/0/1 {
      unit 0 {
        output-traffic-control-profile tc-profile3;
      }
      unit 1 {
        output-traffic-control-profile tc-profile4;
      }
    }
  }
}
```

## 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 3D Universal Edge Router 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 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* 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;
}
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.



**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:

- Adaptive Services PIC, 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
- 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 163](#).

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 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*.

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.



**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 *t1-0/0/0.1*, 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 NxDS0 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 34 on page 183 shows the interfaces/PICs that support fine-grained queuing and scheduling.

**Table 34: Fine-Grained Queuing and Scheduling Support by Interface or PIC Type**

Interface Type	PIC Type	Supported	Example Configuration
<b>IQ PICs</b>			
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;

Table 34: 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 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

Table 34: 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 35 on page 185 shows the MICs and MPCs that support fine-grained queuing and scheduling.

Table 35: Fine-Grained Queuing and Scheduling Support by MIC or MPC Type

MPC	MIC	Supported	Example Configuration
<b>Fixed Configuration MPCs</b>			
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;
<b>MPCs</b>			
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;

Table 35: 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:

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* 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 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*
  - *Example: Applying Scheduling and Shaping to VLANs*
  - *Example: Applying Scheduler Maps and Shaping Rate to DLCIs*

---

## Applying Scheduler Maps to Packet Forwarding Component Queues

On Intelligent Queuing (IQ) and Intelligent Queuing 2 (IQ2) interfaces, 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. To apply a custom chassis scheduler, include the **scheduler-map-chassis** statement at the **[edit class-of-service interfaces at-fpc/pic/\*]** hierarchy level.

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. Include the **scheduler-map-chassis derived** statement at the **[edit class-of-service interfaces type-fpc/pic/\*]** hierarchy level:

```
[edit class-of-service interfaces type-fpc/pic/*]  
scheduler-map-chassis derived;
```



**CAUTION:** If you include 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 include 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. For the Gigabit Ethernet IQ PIC, include both statements.

For more information about the **scheduler-map** statement, see [“Applying Scheduler Maps to Physical Interfaces” on page 158](#). For information about logical interface scheduling configuration, see [“Applying Scheduler Maps and Shaping Rate to DLCIs and VLANs” on page 181](#).

Generally, when you include 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, **ge-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 configuration, 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 190](#). For general information about wildcards, see the *Junos OS Administration Library for Routing Devices*.



**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 189](#)
- [Examples: Scheduling Packet Forwarding Component Queues on page 190](#)

## 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 assign a custom scheduler to the packet forwarding component queues, include the **scheduler-map-chassis** statement at the **[edit class-of-service interfaces type-fpc/pic]** hierarchy level:

```
[edit class-of-service interfaces type-fpc/pic/*]
  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.

For information about defining the scheduler map referenced by **map-name**, see [“Configuring Scheduler Maps” on page 157](#).

## Examples: Scheduling Packet Forwarding Component Queues

### Applying a Chassis Scheduler Map to a 2-Port IQ PIC

Apply a chassis scheduler map to interfaces **ge-0/1/0** and **ge-0/1/1**.

According to customary wildcard rules, the **ge-0/1/0** configuration overrides the **ge-0/1/\*** configuration, implying that the chassis scheduler map **MAP1** is not applied to **ge-0/1/0**. However, the wildcard rule is not obeyed in this case; the chassis scheduler map applies to both interfaces **ge-0/1/0** and **ge-0/1/1**.

```
[edit]
class-of-service {
  interfaces {
    ge-0/1/0 {
      unit 0 {
        classifiers {
          inet-precedence default;
        }
      }
    }
    ge-0/1/* {
      scheduler-map-chassis derived;
    }
  }
}
```

### 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]
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]
interfaces {
  ge-0/1/0 {
    scheduler-map-chassis MAP1;
  }
  ge-0/1/1 {
    scheduler-map-chassis MAP2;
  }
}
```

### Configuring Two T3 Interfaces on a

```
[edit interfaces]
ct3-3/0/0 {
```

```

Channelized DS3 IQ      no-partition interface-type t3; # use entire port 0 as T3
PIC                      }
                          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;
                          }
                          }
                          }
                          }

```

#### Applying Normal Schedulers to Two T3 Interfaces

Configure a scheduler for the aggregated traffic transmitted into both T3 interfaces.

```

[edit class-of-service]
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;
  }
  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;
    }
    ...
}
```

**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/\***.

```
[edit class-of-service]
interfaces {
    t3-3/0/* {
        scheduler-map-chassis derived;
    }
}
```

**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 **ge-3/0/0.0**:

```
[edit class-of-service]
interfaces {
    ge-3/0/* {
        unit 0 {
            scheduler-map MY_SCHED_MAP;
        }
    }
    ge-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 class-of-service]
interfaces {
    ge-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.



## CHAPTER 9

# Setting Transmission Order Using Scheduler Priorities

- [Configuring Schedulers for Priority Scheduling on page 195](#)
- [Associating Schedulers with Fabric Priorities on page 197](#)

### Configuring Schedulers for Priority Scheduling

---

To configure priority scheduling, include the **priority** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level:

```
[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*.

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.

The following sections discuss priority scheduling:

- [Example: Configuring Priority Scheduling on page 195](#)
- [Strict-High Priority Configuration Overview on page 196](#)

### Example: Configuring Priority Scheduling

Configure priority scheduling, as shown in the following example:

1. Configure a scheduler, **be-sched**, with **medium-low** priority.

```
[edit class-of-service]  
schedulers {  
  be-sched {  
    priority medium-low;  
  }  
}
```

```
}
```

2. Configure a scheduler map, **be-map**, that associates **be-sched** with the **best-effort** forwarding class.

```
[edit class-of-service]
scheduler-maps {
  be-map {
    forwarding-class best-effort scheduler be-sched;
  }
}
```

3. Assign **be-map** to a Gigabit Ethernet interface, **ge-0/0/0**.

```
[edit class-of-service]
interfaces {
  ge-0/0/0 {
    scheduler-map be-map;
  }
}
```

## 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 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.



**NOTE:** A queue with **strict-high** priority is assured unlimited transmission bandwidth but is not actually assigned a large delay buffer. Not configuring a **transmit-rate** or an explicit **buffer-size** on a **strict-high** priority queue only ensures that the queue gets assigned a default minimum delay buffer, making it possible, under bursty conditions, to see tail-drops on **strict-high** priority queues. Assigning a small **transmit-rate** or an explicit temporal or percentage **buffer-size** to the queue ensures that the queue has a large enough buffer to hold bursts and protect against tail-drops.

---

**strict-high** priority queues might starve **low** 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. 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, 100 percent of the bandwidth is given to the EF forwarding class because of the **strict-high** setting.

- Related Documentation**
- [Schedulers Overview on page 143](#)
  - [Platform Support for Priority Scheduling](#)

## Associating Schedulers with Fabric Priorities

On Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers 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;
```



**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 161](#)

## CHAPTER 10

# Controlling Congestion Using Scheduler RED Drop Profiles and Buffers

- [Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199](#)
- [RED Drop Profiles Overview on page 200](#)
- [Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 203](#)
- [Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 205](#)
- [Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 206](#)
- [Configuring Rewrite Rules Based on PLP on page 208](#)
- [Example: Configuring Weighted RED Buffer Occupancy on page 208](#)

### Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers

RED drop profiles take action on outgoing packets. When TCM 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 205](#).

By default, the drop profile is mapped to packets with low PLP and any protocol type. To configure how packet loss priorities are mapped to a specified drop profile, include the **drop-profile-map** statement at the **[edit class-of-service schedulers scheduler-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) drop-profile profile-name;
```

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 9](#).



**NOTE:** On Juniper Network MX Series 3D Universal Edge Routers, T4000 Core Routers, EX Series switches, and PTX Series Packet Transport Routers, you can configure only the **any** option for the **protocol** statement.

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

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

To use this drop-profile map, you must configure the settings for the **dp** drop profile at the **[edit class-of-service drop-profiles dp]** hierarchy level. For more information, see [“Managing Congestion Using RED Drop Profiles and Packet Loss Priorities” on page 200](#).

For each scheduler, you can configure separate drop profile maps for each loss priority.

You can configure a maximum of 32 different drop profiles.

**Related  
Documentation**

- [Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 205](#)

---

## RED Drop Profiles Overview

You can configure two parameters to control congestion at the output stage. The first parameter defines the delay-buffer bandwidth, which provides packet buffer space to absorb burst traffic up to the specified duration of delay. Once the specified delay buffer becomes full, packets with 100 percent drop probability are dropped from the head of the buffer. For more information, see [“Managing Congestion on the Egress Interface by Configuring the Scheduler Buffer Size” on page 146](#).

The second parameter defines the drop probabilities across the range of delay-buffer occupancy, supporting the random early detection (RED) process. When the number of packets queued is greater than the ability of the router to empty a queue, the queue

requires a method for determining which packets to drop from the network. To address this, the Junos OS provides the option of enabling RED on individual queues.

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

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

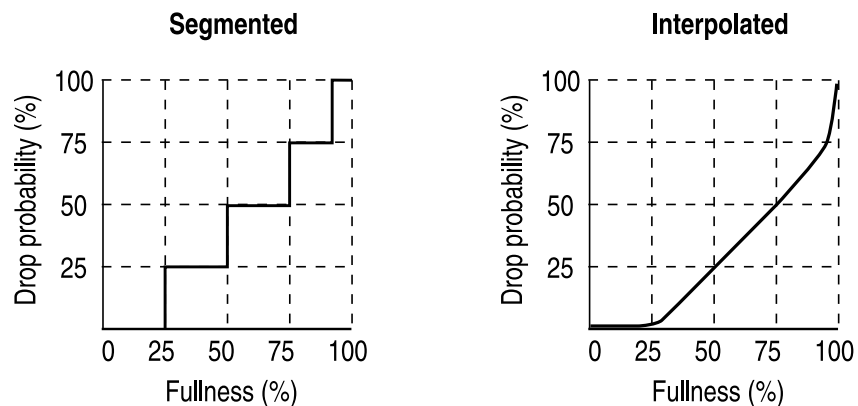
When you configure drop profiles, there are two important values: the queue fullness and the drop probability. The *queue fullness* represents a percentage of the memory used to store packets in relation to the total amount that has been allocated for that specific queue. Similarly, the *drop probability* is a percentage value that correlates to the likelihood that an individual packet is dropped from the network. These two variables are combined in a graph-like format, as shown in [Figure 12 on page 201](#).



**NOTE:** You can only specify two fill levels for interpolated drop profiles on the Enhanced Queuing DPC for Juniper Network MX Series Ethernet Services Routers. For more information about interpolated drop profiles on the Enhanced Queuing DPC for MX Series routers, see *Configuring WRED on Enhanced Queuing DPCs*.

[Figure 12 on page 201](#) shows both a segmented and an interpolated graph. Although the formation of these graph lines is different, the application of the profile is the same. When a packet reaches the head of the queue, a random number between 0 and 100 is calculated by the router. This random number is plotted against the drop profile using the current queue fullness of that particular queue. When the random number falls above the graph line, the packet is transmitted onto the physical media. When the number falls below the graph line, the packet is dropped from the network.

**Figure 12: Segmented and Interpolated Drop Profiles**



By defining multiple fill levels and drop probabilities, you create a segmented drop profile. The line segments are defined in terms of the following graphical model: in the first quadrant, the x axis represents the fill level, and the y axis represents the drop probability. The initial line segment spans from the origin (0,0) to the point ( $\langle l1 \rangle$ ,  $\langle p1 \rangle$ ); a second line runs from ( $\langle l1 \rangle$ ,  $\langle p1 \rangle$ ) to ( $\langle l2 \rangle$ ,  $\langle p2 \rangle$ ) and so forth, until a final line segment connects

(100, 100). The software automatically constructs a drop profile containing 64 fill levels at drop probabilities that approximate the calculated line segments.



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

*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 reference them in each scheduler configuration. For each scheduler, you can configure multiple separate drop profiles, one for each combination of loss priority (low, medium-low, medium-high, or high) and protocol.

You can configure a maximum of 32 different drop profiles.

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

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

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 473](#)
- [drop-probability \(Percentage\)](#)

## 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 26 and [“Overview of Assigning Service Levels to Packets Based on Multiple Packet Header Fields”](#) on page 69.

However, on Juniper Networks M320 Multiservice Edge Routers, MX Series 3D Universal Edge Routers, 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 III Flexible PIC Concentrators (FPCs) and tricolor marking enabled, you can set the PLP with a BA or multifield classifier, as described in [“Using BA Classifiers to Set PLP”](#) on page 35 and [Using Multifield Classifiers to Set Packet Loss Priority](#).

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.

### Example: Overriding the Default PLP on M320 Routers

The following example shows a two-step procedure to override the default PLP settings on M320 routers:

1. The following 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.

```
class-of-service {
  classifiers {
    dscp ba-classifier {
      forwarding-class expedited-forwarding {
        loss-priority high code-points 110;
      }
    }
  }
}
```

2. For M320 routers, this multifield classifier sets the PLP.

```
firewall {
  filter ef-filter {
    term ef-multifield {
      from {
        precedence 6;
      }
      then {
```

```
        loss-priority high;
        forwarding-class expedited-forwarding;
    }
}
}
```

## 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.

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

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

- Related Documentation**
- [Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 205](#)
  - [Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199](#)
  - [Configuring Schedulers on page 146](#)

## Defining Packet Drop Behavior by Configuring RED Drop Profiles

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

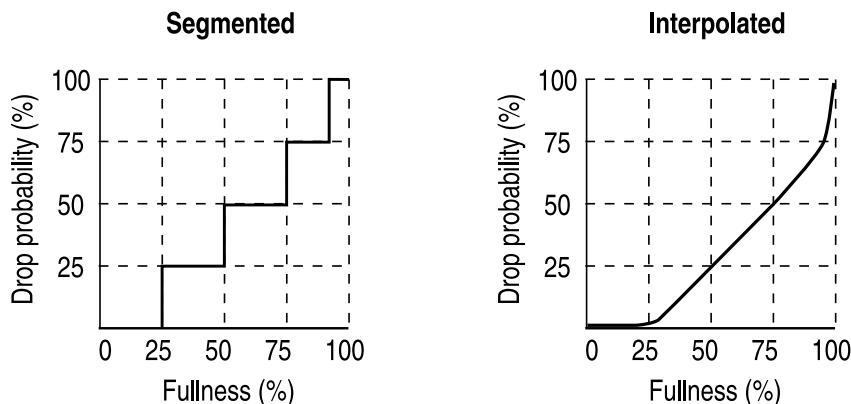
To configure a drop profile, include the **drop-profiles** statement at the **[edit class-of-service]** hierarchy level:

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

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 200](#).

For example, the following shows a segmented configuration and an interpolated configuration that correspond to the graphs in [Figure 13 on page 205](#). The values defined in the configuration are matched to represent the data points in the graph line.

**Figure 13: Segmented and Interpolated Drop Profiles**



1704

**Creating a Segmented Configuration**

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

To create this profile's segmented graph line, the software begins at the bottom-left corner, representing a 0 percent fill level and a 0 percent drop probability. This configuration draws a line directly to the right until it reaches the first defined fill level, 25 percent for this configuration. The software then continues the line vertically until the first drop probability is reached. This process is repeated for all of the defined levels and probabilities until the top-right corner of the graph is reached.

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

**Creating an Interpolated Configuration**

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

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 199.

**Related Documentation**

- [Managing Congestion Using RED Drop Profiles and Packet Loss Priorities](#) on page 200

---

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

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

#### Related Documentation

- [Example: Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 208](#)
- `red-buffer-occupancy`

## 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**, **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 [“Forwarding Classes Overview” on page 111](#).

## Example: Configuring Weighted RED Buffer Occupancy

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

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

```
    }  
  }
```

or

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

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

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

**Related  
Documentation**

- [Managing Transient Traffic Bursts by Configuring Weighted RED Buffer Occupancy on page 206](#)
- *red-buffer-occupancy*



# Altering Outgoing Packet Headers Using Rewrite Rules to Ensure Forwarding Behavior

- [Rewriting Packet Header Information Overview on page 212](#)
- [Applying Default Rewrite Rules on page 213](#)
- [Configuring Rewrite Rules on page 215](#)
- [Applying IEEE 802.1p Rewrite Rules to Dual VLAN Tags on page 216](#)
- [Setting IPv6 DSCP and MPLS EXP Values Independently on page 217](#)
- [Classifiers and Rewrite Rules at the Global and Physical Interface Levels Overview on page 218](#)
- [Applying Rewrite Rules to Output Logical Interfaces on page 218](#)
- [Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 220](#)
- [Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels on page 222](#)
- [Rewriting MPLS and IPv4 Packet Headers on page 224](#)
- [Rewriting the EXP Bits of All Three Labels of an Outgoing Packet on page 228](#)
- [Rewriting IEEE 802.1p Packet Headers with an MPLS EXP Value on page 229](#)
- [Configuring the IEEE 802.1p Field for CoS Host Outbound Traffic on page 231](#)
- [Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic on page 232](#)
- [Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 232](#)
- [Setting Ingress DSCP Bits for Multicast Traffic over Layer 3 VPNs on page 233](#)
- [Example: Per-Node Rewriting of EXP Bits on page 234](#)

## Rewriting Packet Header Information Overview

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.

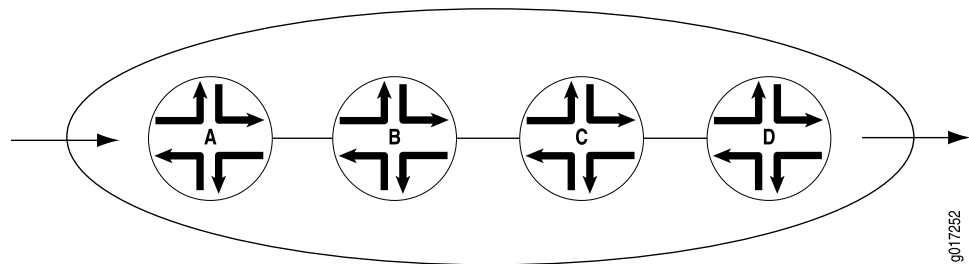
In effect, the rewrite rule performs the opposite function of the behavior aggregate (BA) classifier used when the packet enters the router. As the packet leaves the routing platform, the final CoS action is generally the application of a rewrite rule.

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 router 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 14 on page 212 shows a flow of packets through four routers. 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 14: 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.

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 36 on page 214 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 “Default Aliases for CoS Value Bit Patterns Overview” on page 22, and the default forwarding classes shown in *Default Forwarding Classes*.

When the software detects packets whose CoS values match the forwarding class and PLP values in the first two columns in [Table 36 on page 214](#), the software maps the header bits of those packets to the code-point aliases in the last column in [Table 36 on page 214](#). 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 22](#).

**Table 36: 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 **ge-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 22](#) 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 22](#) 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 22](#) and [Table 36 on page 214](#).

```

class-of-service {
  interfaces {
    ge-1/2/3 {
      unit 0 {
        rewrite-rules {
          dscp default;
        }
      }
    }
  }
}

```

## 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, 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) 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 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 9](#).

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.

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.

On MX Series routers, although you can configure firewall filters and CoS rewrite rules on IRB interfaces, we recommend that you do not configure these functionalities on IRB interfaces because they do not work properly.



**NOTE:** The forwarding class is determined by ingress classification.

**Related Documentation**

- [Applying Rewrite Rules to Output Logical Interfaces](#)
- [Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 232](#)

## 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.

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:

```
class-of-service {
  interfaces {
    ge-0/0/0 {
      unit 0 {
        rewrite-rules {
          ieee-802.1 ieee8021p-rwrule1 vlan-tag outer-and-inner;
        }
      }
    }
  }
}
```

### Setting IPv6 DSCP and MPLS EXP Values Independently

On the M120, M320 with Enhanced III FPCs, MX Series 3D Universal Edge Routers, 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 220](#)

---

## Classifiers and Rewrite Rules at the Global and Physical Interface Levels Overview

On ACX Series Universal Access Routers and EX Series switches, CoS supports classification and rewrite at the global level and physical interface levels.

At a global level, you can define EXP classification.

At a physical interface level, you can define the following features:

- DSCP and inet-precedence classifiers
- DSCP and inet-precedence rewrites
- ieee-802.1 classifiers (inner and outer)
- ieee-802.1 rewrites (outer)

At a logical interface level, you can define the fixed classification and EXP rewrites.

To configure global EXP classifiers, include the **classifiers 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.

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 222](#)

---

## 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;
}
```

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.

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.



**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 EX Series switches.

DSCP rewrite rules are not supported on T Series routers when IQ, IQ2, IQE, IQ2E, 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 217](#)
- [Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 220](#)

---

## 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;
        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.

```
[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 217](#)

---

## Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels

On ACX Series Universal Access 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
        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 and Physical Interface Levels Overview on page 218](#)

## 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.

The default MPLS EXP rewrite table contents are shown in [Table 37 on page 224](#).

**Table 37: Default MPLS EXP Rewrite Table**

Forwarding Class	Loss Priority	CoS 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 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, MX Series routers, and EX Series switches, 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*.

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 **ge-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 **ge-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 **ge-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 **ge-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 **ge-3/1/0**, IPv4 packets transmitted over a non-LSP layer are initialized with values from IP precedence rewrite table **rule2**.
- For interface **ge-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 **ge-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 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 {
  ge-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.
    }
}
}
ge-3/1/1 {
    unit 0 {
        rewrite-rules {
            exp exp-inet-table protocol [mpls mpls-inet-both];
        }
    }
}
}

```

### Example: Simultaneous DSCP and EXP Rewrite

On M120 routers, M320 routers with Enhanced-III FPCs, MX Series routers, and EX Series switches, 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 operations. On these routing devices, 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*. 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 215](#) and [Applying Rewrite Rules to Output Logical Interfaces](#).



**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:

```
[edit class-of-service]
forwarding-classes {
  queue 0 be;
  queue 1 ef;
  queue 2 af;
  queue 3 nc;
}
interfaces {
  ge-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;
    }
  }
}
```

### 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, T Series router, or EX

Series switches, 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.



**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.

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 {
    ge-3/1/0 {
        unit 0 {
            rewrite-rules {
                exp exp-ieee-table;
                ieee-802.1 default;
            }
        }
    }
}
}

```

## 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 3D Universal Edge Routers
- 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 232](#).

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 232](#).

- Related Documentation**
- [Rewriting Packet Headers to Ensure Forwarding Behavior on page 212](#)

## 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 231](#)
  - [Rewriting Packet Headers to Ensure Forwarding Behavior on page 212](#)

## 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 212](#).

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 215](#).
2. Associate the rewrite rules to the desired egress logical interfaces.  
  
See [Applying Rewrite Rules to Output 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).  
  
See [“Overriding the Input Classification” on page 133](#).

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 231](#)
  - [Rewriting Packet Headers to Ensure Forwarding Behavior on page 212](#)

## 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 212](#).

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 Routing Devices*. 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 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;
    }
  }
}
```

---

## 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.

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;
        }
    }
}
```



## PART 3

# Configuring Platform-Specific Functionality

- [Configuring Class of Service on EX Series Ethernet Switches on page 239](#)



# Configuring Class of Service on EX Series Ethernet Switches

- [Classifiers and Rewrite Rules at the Global and Physical Interface Levels Overview on page 239](#)
- [Configuring the Shaping Rate for Physical Interfaces on page 240](#)
- [Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels on page 241](#)

## Classifiers and Rewrite Rules at the Global and Physical Interface Levels Overview

On ACX Series Universal Access Routers and EX Series switches, CoS supports classification and rewrite at the global level and physical interface levels.

At a global level, you can define EXP classification.

At a physical interface level, you can define the following features:

- DSCP and inet-precedence classifiers
- DSCP and inet-precedence rewrites
- ieee-802.1 classifiers (inner and outer)
- ieee-802.1 rewrites (outer)

At a logical interface level, you can define the fixed classification and EXP rewrites.

To configure global EXP classifiers, include the **classifiers 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.

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 222](#)

---

## 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.

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 160](#)

## Configuring Classifiers and Rewrite Rules at the Global and Physical Interface Levels

On ACX Series Universal Access 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
    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 dl;
  }
}
ge-0/1/8 {
  classifiers {
    dscp dl;
  }
}
```

**Related Documentation**

- [Classifiers and Rewrite Rules at the Global and Physical Interface Levels Overview on page 218](#)

## PART 4

# Configuring Line Card-Specific and Interface-Specific Functionality

- [Feature Support for Line Cards and Interfaces on page 245](#)
- [Configuring Class of Service for Tunnels on page 247](#)
- [Configuring Class of Service on IQ and Enhanced IQ \(IQE\) PICs on page 253](#)
- [Configuring Class of Service on Aggregated, Channelized, and Gigabit Ethernet Interfaces on page 263](#)
- [Configuring Class of Service on 10-Gigabit Ethernet LAN/WAN PICs with SFP+ on page 279](#)
- [Configuring Class of Service on MICs, MPCs, and MLCs on page 291](#)



# Feature Support for Line Cards and Interfaces

- [Interface Types That Do Not Support CoS on page 245](#)

## Interface Types That Do Not Support CoS

---

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



**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).
- **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 routing device. This interface is present on the first-hop routing device.

- **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. For a complex configuration example, see the *Junos OS, Release 15.1*.

# Configuring Class of Service for Tunnels

- [CoS for Tunnels Overview on page 247](#)
- [Configuring CoS for Tunnels on page 248](#)
- [Tunneling and BA Classifiers on page 248](#)
- [Example: Configuring CoS for Tunnels on page 249](#)
- [Example: Configuring a GRE Tunnel to Copy ToS Bits to the Outer IP Header on page 252](#)

## CoS for Tunnels Overview

---

For Adaptive Services, 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*. 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);
        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;
}
```

---

## Configuring CoS for Tunnels

To configure CoS for GRE and IP-IP tunnels, perform the following configuration tasks:

1. To configure the tunnel, include the **tunnel** statement at the **[edit interfaces ip-fpc/pic/port unit logical-unit-number]** or **[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** 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.)

---

## 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
- T Series routers

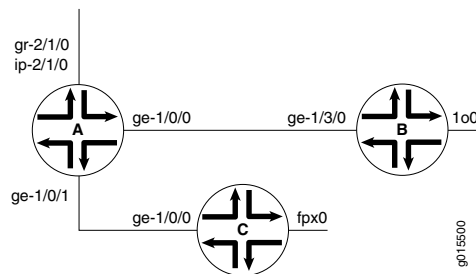
When a GRE or IP-IP tunnel is configured on an incoming (core-facing) interface, the queue number and PLP information are carried through the tunnel. At the egress (customer-facing) interface, the packet is queued and the CoS bits rewritten based on the information carried through the tunnel.

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.

### Example: Configuring CoS for Tunnels

In [Figure 15 on page 249](#), 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 15: CoS with a Tunnel Configuration



```
Router A [edit 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;
    }
  }
}
gr-2/1/0 {
  unit 0 {
    tunnel {
```

```
        source 11.11.11.11;
        destination 10.255.245.46;
    }
    family inet {
        address 21.21.21.21/24;
    }
}
ip-2/1/0 {
    unit 0 {
        tunnel {
            source 12.12.12.12;
            destination 10.255.245.46;
        }
        family inet {
            address 22.22.22.22/24;
        }
    }
}

[edit routing-options]
static {
    route 1.1.1.1/32 next-hop gr-2/1/0.0;
    route 2.2.2.2/32 next-hop ip-2/1/0.0;
}

[edit 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 {
            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;
  }
}

[edit firewall]
filter zf-catch-all {
  term term1 {
    then {
      loss-priority high;
      forwarding-class network-control;
    }
  }
}

```

```

Router B [edit 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;
    }
  }
}

```

```

Router C [edit interfaces]
ge-1/0/0 {
  unit 0 {
    family inet {
      address 10.90.0.1/24;
    }
  }
}

[edit routing-options]

```

```
static {  
    route 1.1.1.1/32 next-hop 10.90.0.2;  
    route 2.2.2.2/32 next-hop 10.90.0.2;  
}
```

---

## Example: 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. This example copies the inner ToS bits to the outer IP header on a GRE tunnel:

```
[edit interfaces]  
gr-0/0/0 {  
    unit 0 {  
        copy-tos-to-outer-ip-header;  
        family inet;  
    }  
}
```

# Configuring Class of Service on IQ and Enhanced IQ (IQE) PICs

- [Simple Filters Overview on page 253](#)
- [Example: Configuring a Simple Filter on page 254](#)
- [BA Classifiers and ToS Translation Tables on page 255](#)
- [CoS for L2TP Tunnels on Ethernet Interface Overview on page 255](#)
- [Configuring CoS for L2TP Tunnels on Ethernet Interfaces on page 256](#)
- [Configuring LNS CoS for Link Redundancy on page 257](#)
- [Example: Configuring L2TP LNS CoS Support for Link Redundancy on page 258](#)

## 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.

## Example: Configuring a Simple Filter

This simple filter sets the loss priority to low for TCP traffic with source address 1.1.1.1, sets the loss priority to high for HTTP (web) traffic with source addresses in the 4.0.0.0/8 range, and sets the loss priority to low for all traffic with destination address 6.6.6.6. The simple filter is applied as an input filter (arriving packets are checking for destination address 6.6.6.6, not queued output packets) on interface **ge-0/0/1.0**.

```
[edit]
firewall {
  family inet {
    simple-filter filter1 {
      term 1 {
        from {
          source-address {
            1.1.1.1/32;
          }
          protocol {
            tcp;
          }
        }
        then loss-priority low;
      }
      term 2 {
        from {
          source-address {
            4.0.0.0/8;
          }
          source-port {
            http;
          }
        }
        then loss-priority high;
      }
      term 3 {
        from {
          destination-address {
            6.6.6.6/32;
          }
        }
        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;
      }
    }
  }
}

```

## 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.

For more information on configuring ToS translation tables, along with examples, see *Configuring ToS Translation Tables*.

## 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.

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).
- 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](#)
- [Configuring LNS CoS for Link Redundancy on page 257](#)
- [Example: Configuring L2TP LNS CoS Support for Link Redundancy on page 258](#)

---

## 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

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](#)
- [CoS for L2TP Tunnels on Ethernet Interface Overview](#)
- [Example: Configuring CoS for L2TP Tunnels on Ethernet Interfaces](#)
- [Configuring LNS CoS for Link Redundancy on page 257](#)
- [Example: Configuring L2TP LNS CoS Support for Link Redundancy on page 258](#)
- `show class-of-service l2tp-session`

## 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

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](#)

---

## 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 258](#)
- [Overview on page 259](#)
- [Configuration on page 259](#)
- [Verification on page 260](#)

## Requirements

Before you begin:

- Configure service and loopback interfaces.
- Configure CoS for L2TP.

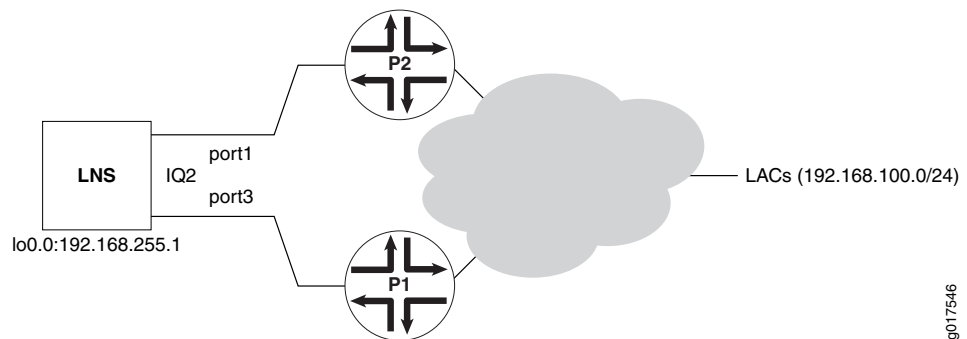
This feature applies to M Series Multiservice Edge Router running Junos OS Release 12.1 or later and EX Series switches.

## 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 16 on page 259 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 16: Topology to Verify Link Redundancy Support for L2TP LNS CoS**



g017546

## 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 ]
```

**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 260](#)
- [Verify LAC Reachability and the Status of L2TP Sessions on page 261](#)

### 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.

**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 257](#)



# Configuring Class of Service on Aggregated, Channelized, and Gigabit Ethernet Interfaces

- [Limitations on CoS for Aggregated Interfaces on page 263](#)
- [Configuring Per-Unit Schedulers for Channelized Interfaces on page 265](#)
- [Configuring Schedulers on Aggregated Interfaces on page 268](#)
- [Policer Support for Aggregated Ethernet Bundle Overview on page 269](#)
- [Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 270](#)
- [Examples: Configuring CoS on Aggregated Interfaces on page 271](#)
- [Example: Configuring Scheduling Modes on Aggregated Interfaces on page 273](#)

## 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 Packet Forwarding Component Queues*, 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 can apply CoS classifiers and rewrite rules directly to the member link interfaces, and the software uses the values you configure.

- You cannot apply a scheduler map to a member link of an aggregate interface.
- 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.

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:

$$\begin{aligned} \text{transmit rate of member link interface} = & \\ & (\text{configured transmit rate of aggregate interface} / \\ & \text{total speed of aggregate interface}) * \\ & (\text{total speed of member link interface} / \text{total configured percent}) * 100 \end{aligned}$$

- 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:

$$\begin{aligned} \text{transmit rate percent of member link interface} = & \\ & (\text{configured transmit rate percent of aggregate interface} / \\ & \text{total configured percent}) * 100 \end{aligned}$$

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.

For more information, see [“Examples: Configuring CoS on Aggregated Interfaces” on page 271](#).

- 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.

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 your 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.

## 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 157](#).



**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 {
    dlc1 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;
  }
  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 {
    unit 0 {
      scheduler-map schedule-mlppp;
    }
  }
  ds-5/3/0:1 {
    unit 0 {
      scheduler-map schedule-mlppp;
    }
  }
  t1-5/3/1:1 {
    unit 0 {
      scheduler-map schedule-mlppp;
    }
  }
}
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;
  }
}
```

## Configuring 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.

To configure CoS schedulers on aggregated interfaces, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
interfaces {
  interface-name {
    scheduler-map map-name;
    unit logical-unit-number {
      scheduler-map map-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;
    priority priority-level;
    transmit-rate (rate | percent percentage | remainder) <exact>;
  }
}
```

## Policer Support for Aggregated Ethernet Bundle Overview

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

You can configure interface-specific policers applied on an aggregated Ethernet bundle 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:

- Interface policers used by the following configuration:  

```
[edit] interfaces (aeX | asX) 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 (aeX | asX) 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 609](#)

---

## Applying Layer 2 Policers to Gigabit Ethernet Interfaces

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

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

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

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

### Examples: Applying Layer 2 Policers to a Gigabit Ethernet Interface

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

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

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

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

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

```
ge-1/0/0 {
  unit 2
  layer2-policer {
```

```

        output-three-color srTCM3-ca { # Apply the srTCM3-color-aware policer.
    }
}

```

## 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;
  }
  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;
    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

---

### 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 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 aen]** 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. For more information about configuring schedulers, see [“Configuring Schedulers” on page 146](#).



**NOTE:** The **traffic-control-profiles** statement is not supported for PTX Series Packet Transport Routers.

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 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.

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**

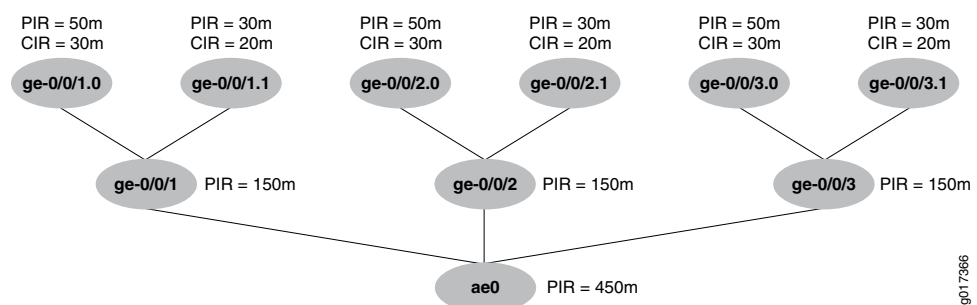
The traffic control profile parameters configured on **ae0.0** are divided across all the underlying logical interfaces (the unit 0s). In the same way, the traffic control profile parameters configured on **ae0.1** are divided across all the underlying logical interfaces (the unit 1s).

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.
- 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 17 on page 276](#).

**Figure 17: 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:
    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.

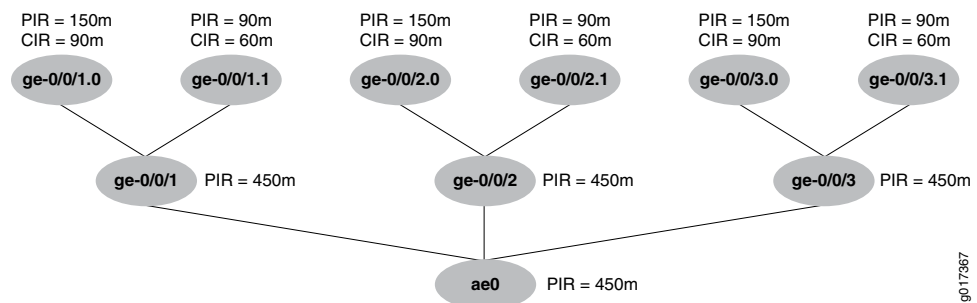
- When the parameters are configured using scheduler maps, then the parameters replicated are the transmit rate and buffer size.

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 18 on page 278](#).

**Figure 18: Replicated Mode for Aggregated Ethernet Interfaces**



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

- [Schedulers Overview on page 143](#)
- [Default Schedulers Overview on page 145](#)
- [Configuring a Scheduler](#)

# 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 279
- DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC with SFP+ on page 280
- BA and Fixed Classification on 10-Gigabit Ethernet LAN/WAN PIC with SFP+ Overview on page 283
- Queuing on 10-Gigabit Ethernet LAN/WAN PICs Properties on page 284
- Scheduling and Shaping on 10-Gigabit Ethernet LAN/WAN PICs Overview on page 285
- Example: Configuring IEEE 802.1p BA Classifier on 10-Gigabit Ethernet LAN/WAN PICs on page 286
- Mapping Forwarding Classes to CoS Queues on 10-Gigabit Ethernet LAN/WAN PICs on page 287
- Example: Configuring Shaping Overhead on 10-Gigabit Ethernet LAN/WAN PICs on page 288
- Configuring DSCP Rewrite for the 10-Gigabit Ethernet LAN/WAN PIC on page 288

## 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

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 38 on page 280](#).

**Table 38: CoS Statements Supported on the 10-Gigabit Ethernet LAN/WAN PICs**

CoS Statements	Supported
<code>buffer-size</code>	No
<code>drop-profile-map</code>	No
<code>excess-priority</code>	No
<code>excess-rate</code>	Yes
<code>priority</code>	Yes
<code>shaping-rate</code>	Yes
<code>transmit-rate</code>	Yes

**Related Documentation**

- *CoS Features and Limitations on M Series and T Series Routers*
- *Junos OS Network Interfaces Library for Routing Devices*

## 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.

**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 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 288](#)
- [dscp on page 396](#)
- [dscp-ipv6 on page 376](#)
- [forwarding-class on page 354](#)
- [rewrite-rules on page 505](#)
- [Understanding DSCP Classification for VPLS](#)
- [Default DSCP and DSCP IPv6 Classifiers on page 42](#)

## 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 32](#).

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 and DSCP classifiers, then the default IP precedence classifier is applied.
- All IPv6 packets are classified using 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.

---

## 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.

**Related Documentation**

- [Mapping Forwarding Classes to CoS Queues on 10-Gigabit Ethernet LAN/WAN PICs on page 287](#)

## 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 39 on page 285](#):

**Table 39: Port Groups on 10-Gigabit Ethernet LAN/WAN PICs**

Port Group	Mapped Ports
Group 1	<code>xe-fpc/pic/0</code>
	<code>xe-fpc/pic/1</code>
Group 2	<code>xe-fpc/pic/2</code>
	<code>xe-fpc/pic/3</code>
Group 3	<code>xe-fpc/pic/4</code>
	<code>xe-fpc/pic/5</code>
Group 4	<code>xe-fpc/pic/6</code>
	<code>xe-fpc/pic/7</code>
Group 5	<code>xe-fpc/pic/8</code>
	<code>xe-fpc/pic/9</code>

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.

---

### 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 {
    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 283](#)

## 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 284](#)
- [Forwarding Classes Overview on page 111](#)

- [Configuring Forwarding Classes on page 116](#)
- [forwarding-classes on page 408](#)

## 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 285](#)

## 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 <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>;
}
```

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 280](#)
- [dscp on page 396](#)
- [dscp-ipv6 on page 376](#)
- [forwarding-class on page 354](#)
- [rewrite-rules on page 505](#)
- [Understanding DSCP Classification for VPLS](#)
- [Default DSCP and DSCP IPv6 Classifiers on page 42](#)



## CHAPTER 18

# Configuring Class of Service on MICs, MPCs, and MLCs

- [CoS Features and Limitations on MIC and MPC Interfaces on page 291](#)
- [Understanding IEEE 802.1p Inheritance push and swap from a Transparent Tag on page 293](#)
- [Configuring IEEE 802.1p Inheritance push and swap from the Transparent Tag on page 294](#)
- [Excess Bandwidth Distribution on MIC and MPC Interfaces Overview on page 296](#)
- [Managing Dedicated and Remaining Queues for Static CoS Configurations on MIC and MPC Interfaces on page 296](#)
- [Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 298](#)
- [CoS on Ethernet Pseudowires in Universal Edge Networks Overview on page 300](#)
- [CoS on Application Services Modular Line Card Overview on page 300](#)
- [Configuring CoS on an Ethernet Pseudowire for Multiservice Edge Networks on page 302](#)
- [Configuring a CoS Scheduling Policy on Logical Tunnel Interfaces on page 304](#)
- [Verifying the Number of Dedicated Queues Configured on MIC and MPC Interfaces on page 305](#)

## CoS Features and Limitations on MIC and MPC Interfaces

MIC and MPC interfaces on MX Series 3D Universal Edge routers 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 40: 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.

Table 40: CoS Limitations on MIC and MPC Interfaces (*continued*)

CoS Feature	Limitation on MIC or MPC Interfaces
BA classifier for MPLS packets	<p>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 <a href="#">“Default MPLS EXP Classifier” on page 49</a>.</p> <p>To enable the default MPLS EXP classifier, include the <b>default</b> statement at the <b>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <a href="#">rewrite-rules exp</a>]</b> hierarchy level.</p>
Rewrite rules	<p>For interfaces on MPCs or on MICs installed in MPCs, you can figure up to 32 rewrite rules:</p> <ul style="list-style-type: none"> <li>• DCSP rewrite rules</li> <li>• Internet rewrite rules</li> <li>• EXP rewrite rules</li> <li>• IEEE rewrite rules</li> </ul> <p>However, if you configure all 32 allowed rewrite rules, the class-of-service process intermittently fails and generates syslog entries.</p>
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 <i>core-facing</i> interface.
Excess bandwidth sharing	<p>Interfaces on MICs and MPCs do not support the <b>excess-bandwidth-share</b> configuration statement, which specifies how excess bandwidth at an interface set in a hierarchical scheduler environment is to be shared: proportionally or equally.</p> <p>Instead, you can include the <b>excess-rate</b> statement at one of the following hierarchy levels:</p> <ul style="list-style-type: none"> <li>• <b>[edit class-of-service <a href="#">schedulers scheduler-name</a>]</b></li> <li>• <b>[edit class-of-service <a href="#">traffic-control-profiles traffic-control-profile-name</a>]</b></li> </ul>
Layer 1 and Layer 2 overhead	<p>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.</p> <p>Queue statistics also take these overheads into account when displaying byte statistics.</p>
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 <b><i>interface-fpc/odd   even/ports</i></b> . For example, if one link is <b>xe-1/0/0</b> , the other should be <b>xe-1/1/0</b> . 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](#)
- [Understanding Hierarchical Scheduling for MIC and MPC Interfaces](#)
- [Scheduler Delay Buffering on MIC and MPC Interfaces](#)

- *Drop Profiles on MIC and MPC Interfaces*

## 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.

### Related Documentation

- *swap-by-poppush*
- [transparent on page 437](#)
- *Understanding swap-by-poppush*
- [Configuring IEEE 802.1p Inheritance push and swap from the Transparent Tag on page 294](#)
- *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;
      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;
        }
      }
    }
  }
}
```

**Related Documentation**

- [transparent on page 437](#)
- *swap-by-poppush*
- [Understanding IEEE 802.1p Inheritance push and swap from a Transparent Tag on page 293](#)
- *Understanding swap-by-poppush*
- *Understanding Transparent Tag Operations and IEEE 802.1p Inheritance*

## 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 298](#)
- *Managing Excess Bandwidth Distribution for Dynamic CoS on MIC and MPC Interfaces*
- *Per-Priority Shaping on MIC and MPC Interfaces Overview*
- *Traffic Burst Management on MIC and MPC Interfaces Overview*

## 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 296](#)
- [Configuring Remaining Common Queues on MIC and MPC Interfaces on page 297](#)

### 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.

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# setmax-queues-per-interface (8 | 4)
```

## 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.

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
```

### Related Documentation

- *Dedicated Queue Scaling for CoS Configurations on MIC and MPC Interfaces Overview*
- [Verifying the Number of Dedicated Queues Configured on MIC and MPC Interfaces on page 305](#)
- *Configuring Hierarchical Schedulers for CoS*
- *Configuring Interface Sets*

## 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.

```
[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.

```
[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 296](#)
- For more information on hierarchical scheduling and operational modes, see *Configuring Hierarchical Schedulers for CoS*.

## 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 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 302](#)

## 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.
- **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.

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 301](#) and [“CoS Implementation in Transparent Proxy Scenario” on page 302](#)

## 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
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 300](#)
- [Rewriting Packet Headers to Ensure Forwarding Behavior on page 212.](#)
- [Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26.](#)

## 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;
}
[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](#)
- [Configuring Hierarchical Schedulers for CoS](#)
- [Configuring Logical Tunnel Interfaces](#)
- [CoS on Ethernet Pseudowires in Universal Edge Networks Overview on page 300](#)
- [Configuring CoS on an Ethernet Pseudowire for Multiservice Edge Networks on page 302](#)

## 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.

**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 296](#)
  - *Managing Dedicated and Remaining Queues for Dynamic CoS Configurations on MIC and MPC Interfaces*

## PART 5

# Configuration Statements

- [Configuration Statement Hierarchies on page 309](#)
- [Configuration Statements: Aggregated Ethernet Interfaces on page 345](#)
- [Configuration Statements: BA Classifiers on page 365](#)
- [Configuration Statements: Code Point Aliases on page 391](#)
- [Configuration Statements: Forwarding Policy Options on page 401](#)
- [Configuration Statements: MIC and MPC Interfaces on page 417](#)
- [Configuration Statements: Packets Using Multifield Classifiers on page 433](#)
- [Configuration Statements: RED Drop Profiles on page 469](#)
- [Configuration Statements: Rewriting Packet Header Information on page 477](#)
- [Configuration Statements: Routing Engine Protocol Queue Assignments on page 509](#)
- [Configuration Statements: Schedulers on page 529](#)
- [Configuration Statements: Tricolor Marking Policers on page 571](#)
- [Configuration Statements: Tunnels CoS on page 631](#)



# Configuration Statement Hierarchies

- [\[edit chassis\] Hierarchy Level on page 309](#)
- [\[edit class-of-service\] Hierarchy Level on page 317](#)
- [\[edit firewall\] Hierarchy Level on page 321](#)
- [\[edit interfaces\] Hierarchy Level on page 333](#)

## [\[edit chassis\] Hierarchy Level](#)

---

```
chassis {
  aggregated-devices {
    ethernet {
      device-count number;
      lacp {
        link-protection {
          non-revertive;
        }
        system-priority;
      }
    }
    sonet {
      device-count number;
    }
    maximum-links maximum-links-limit;
  }
  alarm {
    ds1 {
      ais (ignore | red | yellow);
      ylw (ignore | red | yellow);
    }
    ethernet {
      link-down (ignore | red | yellow);
    }
    integrated-services {
      failure (ignore | red | yellow);
    }
    management-ethernet {
      link-down (ignore | red | yellow);
    }
    relay
    input {
      port port-number {
```

```

        mode (close | open);
        trigger (ignore | red | yellow;
    }
}
output{
    port port-number {
        input-relay input-relay;
        mode (close | open);
        temperature;
    }
}
serial {
    cts-absent (ignore | red | yellow);
    dcd-absent (ignore | red | yellow);
    dsr-absent (ignore | red | yellow);
    loss-of-rx-clock (ignore | red | yellow);
    loss-of-tx-clock (ignore | red | yellow);
    tm-absent (ignore | red | yellow);
}
services {
    hw-down (ignore | red | yellow);
    linkdown (ignore | red | yellow);
    pic-hold-reset (ignore | red | yellow);
    pic-reset (ignore | red | yellow);
    rx-errors (ignore | red | yellow);
    sw-down (ignore | red | yellow);
    tx-errors (ignore | red | yellow);
}
sonet {
    (ais-l | ais-p | ber-sd | ber-sf | locd | lol | lop-p | los | pll | plm-p | rfi-l | rfl-p | uneq-p)
    (ignore | red | yellow);
}
t3 {
    (ais | exz | ferf | idle | lcv | lof | los | pll | ylw) (ignore | red | yellow);
}
}
cluster {
    control-link-recovery;
    control-ports {
        fpc slot-number port port-number;
    }
    heartbeat-interval milliseconds;
    heartbeat-threshold number;
    redundancy-group {
        ... the redundancy-group subhierarchy appears at the end of the [edit chassis cluster]
        hierarchy ...
    }
    reth-count number;
    traceoptions {
        file <filename> <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
        flag flag;
        level severity;
        no-remote-trace;
    }
    redundancy-group group-number {

```

```

    gratuitous-arp-count number;
    hold-down-interval seconds;
    interface-monitor {
        interface-name weight number;
    }
    ip-monitoring {
        family {
            inet {
                ipv4-address {
                    interface rethindex.logical-unit-number secondary-ip-address ipv4-address;
                    weight number;
                }
            }
        }
        global-threshold number;
        global-weight number;
        retry-count count;
        retry-interval interval;
    }
    node node-number priority priority-number;
    preempt;
}
config-button {
    no-clear;
    no-rescue;
}
container-devices {
    device-count number;
}
craft-lockout;
disable-power-management;
disk-partition partition-name (/config | /var) {
    level (full | high) {
        free-space threshold-value (mb | percent);
    }
}
enhanced-policer;
extended-statistics;
fabric {
    degraded {
        action-fpc-restart-disable;
        degraded-fabric-detection-enable
        degraded-fpc-bad-plane-threshold number-bad-planes;
    }
    redundancy-mode (increased-bandwidth | redundant);
}
filter;
fpc slot-number {
    ... the fpc subhierarchy appears after the main [edit chassis] hierarchy ...
}
fpc-feb-connectivity {
    fpc slot-number feb (slot-number | none);
}
fpc-resync;
fru-poweron-sequence sequence;
lcc index {

```

```

... the lcc subhierarchy appears after the main [edit chassis] hierarchy ...
}
maximum-ecmp value;
memory-enhanced {
    filter;
    route;
    vpn-label;
}
network-services (ethernet | enhanced-ethernet | ip | enhanced-ip);
(packet-scheduling | no-packet-scheduling);
pem {
    minimum number;
}
policer-drop-probability-low;
ppp-subscriber-services (disable | enable);
redundancy {
    cfeb slot (always | preferred);
    failover {
        on-disk-failure;
        on-loss-of-keepalives;
    }
    feb {
        redundancy-group group-name {
            description description;
            feb slot-number <backup | primary>;
            no-auto-failover;
        }
    }
    graceful-switchover;
    keepalive-time seconds;
    routing-engine slot-number (backup | disabled | master);
    sfm slot-number (always | preferred);
    ssb slot-number (always | preferred);
}
route-localization {
    inet (chassis);
    inet6;
}
routing-engine {
    bios {
        no-auto-upgrade;
    }
    on-disk-failure disk-failure-action (halt | reboot);
}
sfm slot-number {
    power off;
}
sib {
    minimum number;
}
(source-route | no-source-route);
state [
    cb-upgrade [on | off];
]
synchronization { # for M Series and T Series routers
    primary (external-a | external-b);
}

```

```

secondary (external-a | external-b);
signal-type (e1 | t1);
switching-mode (non-revertive | revertive);
transmitter-enable;
validation-interval seconds;
y-cable-line-termination;
}
synchronization { # for MX80 and MX240 routers
  clock-mode (auto-select | free-run);
  esmc-transmit {
    interfaces (all | interface-name);
  }
  hold-interval {
    configuration-change seconds;
    restart seconds;
    switchover seconds;
  }
  network-option (option-1 | option-2);
  quality-mode-enable;
  selection-mode (configured-quality|received-quality);
  source {
    (external-a | external-b) {
      priority number;
      quality-level (prc | prs |sec | smc | ssu-a | ssu-b | st2 | st3 | st3e | st4 | stu | tnc);
      request (force-switch | lockout);
    }
    interfaces interface-name {
      priority number;
      quality-level (prc | prs |sec | smc | ssu-a | ssu-b | st2 | st3 | st3e | st4 | stu | tnc);
      request (force-switch | lockout);
      wait-to-restore minutes;
    }
  }
  switchover-mode (revertive | non-revertive);
}
synchronization { # for ACX Series routers
  clock-mode (auto-select | free-run);
  esmc-transmit {
    interfaces (all | interface-name);
  }
  hold-interval {
    configuration-change seconds;
    restart seconds;
    switchover seconds;
  }
  network-option (option-1 | option-2);
  quality-mode-enable;
  selection-mode (configured-quality | received-quality);
  source {
    (bits | gps) {
      priority number;
      quality-level (prc | prs |sec | smc | ssu-a | ssu-b | st2 | st3 | st3e | st4 | stu | tnc);
      request (force-switch | lockout);
    }
    interfaces interface-name {
      priority number;

```

```

        quality-level (prc | prs | sec | smc | ssu-a | ssu-b | st2 | st3 | st3e | st4 | stu | tnc);
        request (force-switch | lockout);
        wait-to-restore minutes;
    }
}
switchover-mode(non-revertive | revertive);
}
system-domains {
    protected-system-domains psdnumerical-index {
        control-plane-bandwidth-percent percent;
        control-slot-numbers [ slot-numbers ];
        control-system-id control-system-id;
        description description;
        fpcs [ slot-numbers ];
    }
    root-domain-id root-domain-id;
}
vrf-mtu-check;
}

chassis {
    fpc slot-number {
        number-of-ports active-ports;
        offline;
        pic slot-number {
            ... the pic subhierarchy appears after the main [edit chassis fpc slot-number] hierarchy
            ...
        }
        port-mirror-instance port-mirror-instance-name;
        power (off | on);
        sampling-instance instance-name;
    }
}

fpc slot-number {
    pic slot-number {
        adaptive-services {
            service-package (layer-2 | layer-3 | ...the following extension-provider subhierarchy
            ...);
            extension-provider {
                control-cores number;
                data-cores number;
                data-flow-affinity {
                    hash-key (layer-3 | layer-4);
                }
                channelization;
                forwarding-db-size megabytes;
                object-cache-size megabytes;
                package package-name;
                policy-db-size megabytes;
                syslog {
                    facility {
                        severity;
                        destination (pic-console | routing-engine);
                    }
                }
            }
            wired-process-mem-size megabytes;
        }
    }
}

```

```

    }
  }
  aggregated-devices {
    ima {
      device-count number;
    }
  }
  aggregate-ports;
  atm-cell-relay-accumulation;
  atm-l2circuit-mode (aal5 | cell | trunk trunk);
  cel {
    e1 port-number {
      channel-group group-number timeslots slot-number;
    }
  }
  ct3 {
    port port-number {
      t1 link-number {
        channel-group group-number timeslots slot-number;
      }
    }
  }
  ethernet {
    pic-mode (enhanced-switching | routing | switching);
  }
  fibre-channel {
    port port-number;
    port-range port-range-low port-range-high
  }
  egress-policer-overhead bytes;
  forwarding-mode {
    sa-multicast;
    vlan-steering {
      vlan-rule (high-low | odd-even);
    }
  }
  framing (e1 | e3 | sdh | sonet | t1 | t3);
  idle-cell-format {
    itu-t;
    payload-pattern payload-pattern-byte;
  }
  ingress-policer-overhead bytes;
  inline-services {
    bandwidth (1g | 10g);
  }
  linerate-mode;
  max-queues-per-interface (4 | 8);
  mlfr-uni-nni-bundles number;
  no-concatenate;
  no-multi-rate;
  port port-number {
    framing (e1 | e3 | sdh | sonet | t1 | t3);
    forwarding-mode {
      sa-multicast;
    }
    speed ( oc3-stm1 | oc12-stm4 | oc48-stm16);
  }

```

```

    }
    port-mirror-instance port-mirror-instance-name;
    q-pic-large-buffer {
        (large-scale | small-scale);
    }
    red-buffer-occupancy {
        weighted-averaged <instant-usage-weight-exponent weight-value>;
    }
    shdsl {
        pic-mode (1-port-atm | 2-port-atm);
    }
    sparse-dlcis;
    traffic-manager {
        egress-shaping-overhead number;
        ingress-shaping-overhead number;
        mode {
            egress-only;
            ingress-and-egress;
            session-shaping;
        }
    }
    tunnel-queuing;
    tunnel-services {
        bandwidth (1g | 10g | 20g | 40g);
        tunnel-only;
    }
    vtmapping (itu-t | klm);
}
}

chassis {
    lcc index {
        fpc slot-number {
            ... the fpc subhierarchy appears after the main [edit chassis lcc index] hierarchy ...
        }
        offline;
        online-expected;
    }
}

lcc index {
    fpc slot-number {
        pic slot-number {
            ... the pic subhierarchy appears after the main [edit chassis lcc index fpc slot-number] hierarchy ...
        }
        power (off | on);
        sampling-instance instance-name;
    }

    fpc slot-number {
        pic slot-number {
            aggregate-ports;
            atm-cell-relay-accumulation;
            atm-l2circuit-mode (aal5 | cell | trunk trunk);

```



```

        fill-level percentage drop-probability percentage;
        interpolate {
            drop-probability value;
            fill-level value;
        }
    }
}
fabric {
    scheduler-map {
        priority (high | low) scheduler scheduler-name;
    }
}
forwarding-class-map {
    map-name {
        class class-name queue-num queue-number <restricted-queue queue-number>;
    }
}
forwarding-classes {
    class class-name policing-priority (normal | premium) queue-num queue-number
        priority (high | low);
    queue queue-number class-name policing-priority (normal | premium) priority (high |
        low);
}
forwarding-policy {
    class class-name {
        classification-override {
            forwarding-class class-name;
        }
    }
}
next-hop-map map-name {
    forwarding-class class-name {
        discard;
        lsp-next-hop [ lsp-regular-expressions ];
        next-hop [ next-hop-names ];
        non-lsp-next-hop;
    }
}
}
fragmentation-maps {
    map-name {
        forwarding-class class-name {
            drop-timeout milliseconds;
            fragment-threshold bytes;
            multilink-class number;
            no-fragmentation;
        }
    }
}
}
host-outbound-traffic {
    dscp-code-point value;
    forwarding-class class-name;
    ieee-802.1 {
        default value;
        rewrite-rules;
    }
}
tcp {

```

```

        raise-internet-control-priority;
    }
}
interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
}
restricted-queues {
    forwarding-class class-name queue-number;
}
rewrite-rules {
    (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
    rewrite-rule {
        forwarding-class class-name {
            loss-priority level code-point (alias | bits);
        }
        import (rewrite-rule | default);
    }
}
routing-instances routing-instance-name {
    classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
    }
}
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}
schedulers {
    scheduler-name {
        adjust-minimum value;
        adjust-percent value;
        buffer-size (exact | percent percentage | remainder);
        drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
            protocol any;
        excess-priority (high | low | medium-high | medium-low);
        excess-rate (percent percentage | proportion proportion);
        priority (high | low | medium-high | medium-low | strict-high);
        shaping-rate (bps | percent percentage | burst-size size);
        transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
    }
}
traceoptions {
    file <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
}
traffic-control-profiles {
    profile-name {
        adjust-minimum rate;
    }
}

```

```

    delay-buffer-rate (bps | cps cps | percent percentage);
    excess-rate (percent percentage | proportion value);
    guaranteed-rate (bps | percent percentage) <burst-size bytes>;
    overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
    scheduler-map map-name;
    shaping-rate (bps | percent percentage) <burst-size bytes>;
  }
}
tri-color;
}

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

```

```

        ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        inet-precedence (rewrite-name | default) <protocol mpls>;
    }
    scheduler-map map-name;
    shaping-rate bps;
    translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
        to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
    }
}
interface-set interface-set-name {
    excess-bandwidth-share (equal | proportional value);
    input-excess-bandwidth-share (equal | proportional value);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
}
}
}

```

**Related  
Documentation**

- [Notational Conventions Used in Junos OS Configuration Hierarchies](#)

## [\[edit firewall\] Hierarchy Level](#)

Several statements in the **[edit firewall]** hierarchy are valid at numerous locations within the hierarchy. .

- [Common Firewall Actions on page 321](#)
- [Common IP Firewall Actions on page 322](#)
- [Common IPv4 and IPv6 Firewall Actions on page 322](#)
- [Common IP Firewall Match Conditions on page 323](#)
- [Common IPv4 Firewall Match Conditions on page 324](#)
- [Common Layer 2 Firewall Match Conditions on page 324](#)
- [Complete \[edit firewall\] Hierarchy on page 326](#)

## Common Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on [page 326](#) instead of the statements being repeated.

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

The common firewall actions are as follows:

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

## Common IP Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on page 326 instead of the statements being repeated.

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

The common IP firewall actions are as follows:

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

## Common IPv4 and IPv6 Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on page 326 instead of the statements being repeated.

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

The common IP version 4 (IPv4) and IP version 6 (IPv6) firewall actions are as follows:

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

## Common IP Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on page 326 instead of the statements being repeated.

- **[edit firewall family inet dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in “[Complete \[edit firewall\] Hierarchy](#)” on page 326)
- **[edit firewall family inet filter *filter-name* term *term-name* from]**
- **[edit firewall family inet6 dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in “[Complete \[edit firewall\] Hierarchy](#)” on page 326)
- **[edit firewall family inet6 filter *filter-name* term *term-name* from]**
- **[edit firewall filter *filter-name* term *term-name* from]**

The common IP firewall match conditions are as follows:

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

## Common IPv4 Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in [“Complete \[edit firewall\] Hierarchy” on page 326](#) instead of the statements being repeated.

- **[edit firewall family inet dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in [“Complete \[edit firewall\] Hierarchy” on page 326](#))
- **[edit firewall family inet filter *filter-name* term *term-name* from]**
- **[edit firewall filter *filter-name* term *term-name* from]**

The common IPv4 firewall match conditions are as follows:

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

## Common Layer 2 Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in [“Complete \[edit firewall\] Hierarchy” on page 326](#) instead of the statements being repeated.

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

The common Layer 2 firewall match conditions are as follows:

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

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

## Complete [edit firewall] Hierarchy

```

firewall {
  family (any | ethernet-switching | inet | inet6) {
    ... the family subhierarchies appear after the main [edit firewall] hierarchy ...
  }
  filter filter-name {
    accounting-profile [ profile-names ];
    enhanced-mode;
    interface-shared-with;
    interface-specific;
    physical-interface-policer;
    term term-name {
      filter filter-name;
      from {
        ... statements in Common IP Firewall Match Conditions on page 323 AND
        statements in Common IPv4 Firewall Match Conditions on page 324 ...
      }
      then {
        ... statements in Common Firewall Actions on page 321 AND
        statements in Common IP Firewall Actions on page 322 AND
        statements in Common IPv4 and IPv6 Firewall Actions on page 322 ...
      }
    }
  }
}
hierarchical-policer policer-name {
  aggregate {
    if-exceeding {
      bandwidth-limit bps;
      burst-size-limit bytes;
    }
    then {
      discard;
      forwarding-class class-name;
      loss-priority (high | low | medium-high | medium-low);
    }
  }
  logical-interface-policer;
  physical-interface-policer;
  premium {
    if-exceeding {
      bandwidth-limit bps;
      burst-size-limit bytes;
    }
    then {
      discard;
    }
  }
}
shared-bandwidth-policer;
interface-set interface-set-name {
  interface-name;
}
load-balance-group group-name {
  next-hop-group [ group-names ];
}

```

```

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

firewall {
  family any {
    filter filter-name {
      interface-shared;
      term term-name {
        from {
          (forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
          interface interface-name;
          interface-set set-name;
          (loss-priority [ priorities ] | loss-priority-except [ priorities ]);
          (packet-length [ values ] | packet-length-except [ values ]);
        }
        then {
          ... statements in Common Firewall Actions on page 321 PLUS ...
          (accept | discard);
        }
      }
    }
  }
}

```

```
    }  
  }  
}
```

```
firewall {  
  family ethernet-switching {  
    filter filter-name {  
      interface-specific;  
      term term-name {  
        from {  
          destination-address {  
            ip-prefix</prefix-length>;  
          }  
          destination-mac-address {  
            mac-address;  
          }  
          destination-port [ port-names ];  
          destination-prefix-list {  
            list-name;  
          }  
          dot1q-tag [ tag-values ];  
          dot1q-user-priority [ priority-values ];  
          dscp [ code-point-values ];  
          ether-type [ protocol-names ];  
          fragment-flags flag;  
          icmp-code [ codes ];  
          icmp-type [ types ];  
          interface interface-name;  
          is-fragment;  
          precedence [ precedence-names ];  
          protocol [ protocol-names ];  
          source-address {  
            ip-prefix</prefix-length>;  
          }  
          source-mac-address {  
            mac-address;  
          }  
          source-port [ port-names ];  
          source-prefix-list {  
            list-name;  
          }  
          tcp-established;  
          tcp-flags flag;  
          tcp-initial;  
          vlan [ vlan-names ];  
        }  
        then {  
          (accept | discard);  
          analyzer analyzer-name;  
          count counter-name;  
          forwarding-class class-name;  
          interface interface-name;  
          log;  
        }  
      }  
    }  
  }  
}
```

```

        loss-priority (high | low);
        policer policer-name;
        syslog;
        vlan vlan-name;
    }
}
}
}
}

firewall {
    family inet {
        dialer-filter filter-name {
            accounting-profile [ profile-names ];
            term term-name {
                from {
                    ... statements in Common IP Firewall Match Conditions on page 323 AND
                    statements in Common IPv4 Firewall Match Conditions on page 324 EXCEPT
                    FOR ...
                    (ah-spi [ values ] | ah-spi-except [ values ]); # NOT valid at this level
                    (destination-class [ class-names ] |
                     destination-class-except [ class-names ]); # NOT valid at this level
                    interface interface-name; # NOT valid at this level
                    (loss-priority [ priorities ] | loss-priority-except [ priorities ]); # NOT valid at
                     this level
                    service-filter-hit; # NOT valid at this level
                    (source-class [ class-names ] | source-class-except [ class-names ]); # NOT
                     valid at this level
                }
                then {
                    (ignore | note);
                    log;
                    sample;
                    syslog;
                }
            }
        }
    }
    filter filter-name {
        accounting-profile [ profile-names ];
        interface-specific;
        term term-name {
            filter filter-name;
            from {
                ... statements in Common IP Firewall Match Conditions on page 323 AND
                statements in Common IPv4 Firewall Match Conditions on page 324 ...
            }
            then {
                ... statements in Common Firewall Actions on page 321 AND
                statements in Common IP Firewall Actions on page 322 AND
                statements in Common IPv4 and IPv6 Firewall Actions on page 322 ...
            }
        }
    }
    prefix-action name {
        count;
        destination-prefix-length prefix-length;
    }
}

```

```

filter-specific;
policer policer-name;
source-prefix-length prefix-length;
subnet-prefix-length prefix-length;
}
service-filter filter-name {
  term term-name {
    from {
      address {
        ip-prefix</prefix-length>;
      }
      (ah-spi [ values ] | ah-spi-except [ values ]);
      destination-address {
        ip-prefix</prefix-length>;
      }
      (destination-port [ port-names ] | destination-port-except [ port-names ]);
      destination-prefix-list {
        list-name;
      }
      (esp-spi [ values ] | esp-spi-except [ values ]);
      first-fragment;
      fragment-flags flag;
      (fragment-offset [ offsets ] | fragment-offset-except [ offsets ]);
      (interface-group [ group-names ] | interface-group-except [ group-names ]);
      (ip-options [ option-names ] | ip-options-except [ option-names ]);
      is-fragment;
      (loss-priority [ priorities ] | loss-priority-except [ priorities ]);
      (port [ port-names ] | port-except [ port-names ]);
      prefix-list {
        list-name;
      }
      (protocol [ protocol-names ] | protocol-except [ protocol-names ]);
      source-address {
        ip-prefix</prefix-length>;
      }
      (source-port [ port-names ] | source-port-except [ port-names ]);
      source-prefix-list {
        list-name;
      }
      tcp-flags flag-name;
    }
    then {
      count counter-name;
      log;
      port-mirror;
      sample;
      (service | skip);
    }
  }
}
simple-filter filter-name {
  term term-name {
    from {
      destination-address ip-prefix</prefix-length>;
      destination-port port-name;
      forwarding-class [ class-names ];
    }
  }
}

```

```

        protocol protocol-name;
        source-address ip-prefix</prefix-length>;
        source-port port-name;
    }
    then {
        forwarding-class class-name;
        loss-priority (high | low | medium-high | medium-low);
        policer policer-name;
    }
}
}
}
}

firewall {
    family inet6 {
        dialer-filter filter-name {
            accounting-profile [ profile-names ];
            term term-name {
                from {
                    ... statements in Common IP Firewall Match Conditions on page 323 PLUS ...
                    (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
                    ... BUT NOT ...
                    (destination-class [ class-names ] |
                     destination-class-except [ class-names ]); # NOT valid at this level
                    (forwarding-class [ class-names ] |
                     forwarding-class-except [ class-names ]); # NOT valid at this level
                    interface interface-name; # NOT valid at this level
                    (interface-group [ group-names ] | interface-group-except [ group-names ]); #
                     NOT valid at this level
                    (loss-priority [ priorities ] | loss-priority-except [ priorities ]); # NOT valid at
                     this level
                    service-filter-hit; # NOT valid at this level
                    (source-class [ class-names ] | source-class-except [ class-names ]); # NOT
                     valid at this level
                    tcp-established; # NOT valid at this level
                    tcp-flags flag; # NOT valid at this level
                    tcp-initial; # NOT valid at this level
                }
                then {
                    (ignore | note);
                    log;
                    sample;
                    syslog;
                }
            }
        }
    }
    filter filter-name {
        accounting-profile [ profile-names ];
        interface-specific;
        term term-name {
            filter filter-name;
            from {
                ... statements in Common IP Firewall Match Conditions on page 323 PLUS ...
                (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
                (traffic-class [ code-point-values ] | traffic-class-except [ code-point-values ]);
            }
        }
    }
}

```

```

    }
    then {
        ... statements in Common Firewall Actions on page 321 AND
        statements in Common IP Firewall Actions on page 322 PLUS ...
        (accept | discard | reject <address-unreachable | administratively-prohibited |
        beyond-scope | fragmentation-needed | no-route | port-unreachable |
        tcp-reset>);
    }
}
}
service-filter filter-name {
    term term-name {
        from {
            address {
                ip-prefix </prefix-length>;
            }
            (ah-spi [ values ] | ah-spi-except [ values ]);
            destination-address {
                ip-prefix </prefix-length>;
            }
            (destination-port [ port-names ] | destination-port-except [ port-names ]);
            destination-prefix-list {
                list-name;
            }
            (esp-spi [ values ] | esp-spi-except [ values ]);
            (interface-group [ group-names ] | interface-group-except [ group-names ]);
            (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
            (port [ port-names ] | port-except [ port-names ]);
            prefix-list {
                list-name;
            }
            source-address {
                ip-prefix </prefix-length>;
            }
            (source-port [ port-names ] | source-port-except [ port-names ]);
            source-prefix-list {
                list-name;
            }
            tcp-flags flag-name;
        }
        then {
            count counter-name;
            log;
            port-mirror;
            sample;
            (service | skip);
        }
    }
}
}
}

```

Related Documentation • *Notational Conventions Used in Junos OS Configuration Hierarchies*

## [edit interfaces] Hierarchy Level

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

```

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

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

    traceoptions {
      flag flag;
    }
    (traps | no-traps);
    unit logical-unit-number {
      accounting-profile name;
      bandwidth rate;
      description text;
      disable;
      encapsulation type;
      family inet {
        accounting {
          destination-class-usage;
          source-class-usage {
            input;
            output;
          }
        }
      }
      address ipv4-address {
        arp ip-address (mac | multicast-mac) mac-address <publish>;
        broadcast address;
        preferred;
        primary;
        vrrp-group group-id {
          (accept-data | no-accept-data);
          advertise-interval seconds;
          advertisements-threshold number;
        }
      }
    }
  }
}

```

```

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

```

```

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

```

```

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

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

```

```

        start-time time;
        interval number;
    }
    source-address-filter {
        mac-address;
    }
    (source-filtering | no-source-filtering);
}
auto-configure {
    remove-when-no-subscribers;
    stacked-vlan-ranges {
        access-profile profile-name;
        authentication {
            password password-string;
            username-include {
                circuit-type;
                delimiter delimiter-character;
                domain-name domain-name-string;
                interface-name;
                mac-address;
                option-82 ( circuit-id | remote-id);
                radius-realm radius-realm-string;
                user-prefix user-prefix-string;
            }
        }
        dynamic-profile profile-name {
            accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
            ranges (any | low-tag-high-tag), (any | low-tag-high-tag);
        }
    }
}
vlan-ranges {
    access-profile profile-name;
    authentication {
        password password-string;
        username-include {
            circuit-type;
            delimiter delimiter-character;
            domain-name domain-name-string;
            interface-name;
            mac-address;
            option-82;
            radius-realm radius-realm-string;
            user-prefix user-prefix-string;
        }
    }
    dynamic-profile profile-name {
        accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
        ranges (any | low-tag)—(any | high-tag);
    }
}
override tag vlan-tag dynamic-profile profile name;
}
encapsulation (ethernet-bridge | ethernet-vpls | extended-vlan-bridge |
    extended-vlan-vpls | flexible-ethernet-services | vlan-vpls);
ether-options {
    802.3ad {

```

```

aex;
(backup | primary);
lacp {
    force-up;
    port-priority
}
}
asynchronous-notification;
(auto-negotiation | no-auto-negotiation);
ethernet-switch-profile {
    ethernet-policer-profile {
        input-priority-map {
            ieee802.1p premium [ values ];
        }
        output-priority-map {
            classifier {
                premium {
                    forwarding-class class-name {
                        loss-priority (high | low);
                    }
                }
            }
        }
        policer cos-policer-name {
            aggregate {
                bandwidth-limit bps;
                burst-size-limit bytes;
            }
            premium {
                bandwidth-limit bps;
                burst-size-limit bytes;
            }
        }
    }
    tag-protocol-id;
}
(mac-learn-enable | no-mac-learn-enable);
}
(flow-control | no-flow-control);
ignore-l3-incompletes;
link-mode (automatic | full-duplex | half-duplex);
(loopback | no-loopback);
keepalives <interval seconds> <down-count number> <up-count number>;
speed (1g | 10m | 100m | 10m-100m | auto-negotiation);
source-address-filter {
    mac-address;
}
source-filtering | no-source-filtering;
}
flexible-vlan-tagging;
(gratuitous-arp-reply | no-gratuitous-arp-reply);
hold-time (up milliseconds | down milliseconds);
interface-transmit-statistics;
(keepalives <down-count number> <interval seconds> <up-count number> |
no-keepalives);
layer2-policer {
    apply-groups [ group-names ];

```

```

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

interface-name {
    unit logical-unit-number {
        disable;
        accept-source-mac {
            mac-address mac-address {
                policer {
                    input policer-name;
                    output policer-name;
                }
            }
        }
    }
    accounting-profile name;
    advisory-options {
        downstream-rate rate;
        upstream-rate rate;
    }
    arp-resp (restricted|unrestricted);
    bandwidth rate;
    clear-dont-fragment-bit;
}

```

```

copy-tos-to-outer-ip-header;
demux-destination family;
encapsulation (vlan-bridge | vlan-vpls);
epd-threshold cells plp1 cells;
filter filter-name;
inner-vlan-id-range start start-id end end-id;
input-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
    inner-tag-protocol-id tpid;
    inner-vlan-id number;
    tag-protocol-id tpid;
    vlan-id number;
}
interface-shared-with psd numerical-index;
layer2-policer {
    input-hierarchical-policer policer-name;
    input-policer policer-name;
    input-three-color policer-name;
    output-policer policer-name;
    output-three-color policer-name;
}
multi-chassis-protection peer-ip-address {
    interface interface-name;
}
native-inner-vlan-id number;
output-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
    inner-tag-protocol-id tpid;
    inner-vlan-id number;
    tag-protocol-id tpid;
    vlan-id number;
}
peer-interface interface-name;
peer-unit unit-number;
plp-to-clp;
proxy-arp <restricted | unrestricted>;
rpm {
    (client | server);
    twamp-server;
}
swap-by-poppush;
vlan-id number;
vlan-id-list [ vlan-id vlan-id-vlan-id ];
vlan-id-range number-number;
vlan-tags (inner <tpid.>vlan-id | inner-list [ vlan-id vlan-id-vlan-id ] |
    inner-range <tpid.>vlan-id-vlan-id) outer <tpid.>vlan-id;
}

unit logical-unit-number {
    family ethernet-switching {
        filter {
            group filter-group-number;
            (input filter-name | input-list [ filter-names ]);
        }
    }
}

```

```

(output filter-name | output-list [ filter-names ]);
(inner-vlan-id-list [ vlan-ids ] | vlan-id number | vlan-id-list [ number
    number-number ]);
interface-mode (access | trunk);
policer {
    input policer-name;
    output policer-name;
}
vlan-rewrite {
    translate old-vlan-id new-vlan-id;
}
vlan {
    members [ all vlan-identifiers ];
}
}
family inet {
    filter {
        group filter-group-number;
        (input filter-name | input-list [ filter-names ]);
        (output filter-name | output-list [ filter-names ]);
    }
    input-hierarchical-policer policer-name;
    mac-validate (loose | strict);
    mtu bytes;
    no-neighbor-learn;
    no-redirects;
    policer {
        arp policer-template-name;
        input policer-name;
        output policer-name;
    }
    primary;
    receive-options-packets;
    receive-ttl-exceeded;
    rpf-check {
        fail-filter filter-name;
        mode loose;
    }
    sampling {
        (input | output | input output);
    }
    simple-filter {
        input filter-name;
    }
    targeted-broadcast {
        forward-and-send-to-re;
        forward-only;
    }
    unnumbered-address interface-name <destination address>
        <destination-profile profile-name> <preferred-source-address address>;
}
}
family inet6 {
    address ipv6-address {
        destination destination-address;

```

```

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

```

```

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

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

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

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

```

#### Related Documentation

- *Notational Conventions Used in Junos OS Configuration Hierarchies*



# Configuration Statements: Aggregated Ethernet Interfaces

- [\[edit class-of-service\] Hierarchy Level](#) on page 345
- [buffer-size \(Schedulers\)](#) on page 350
- [drop-profile \(Schedulers\)](#) on page 351
- [drop-profile-map \(Schedulers\)](#) on page 351
- [excess-priority](#) on page 352
- [excess-rate](#) on page 353
- [forwarding-class \(Interfaces\)](#) on page 354
- [interfaces](#) on page 355
- [loss-priority \(Scheduler Drop Profiles\)](#) on page 357
- [priority \(Schedulers\)](#) on page 358
- [protocol \(Schedulers\)](#) on page 359
- [scheduler-map \(Interfaces and Traffic-Control Profiles\)](#) on page 360
- [scheduler-maps \(For Most Interface Types\)](#) on page 360
- [schedulers \(CoS\)](#) on page 361
- [transmit-rate \(Schedulers\)](#) on page 362
- [unit](#) on page 364

## [\[edit class-of-service\] Hierarchy Level](#)

---

```
class-of-service {
  adaptive-shapers {
    adaptive-shaper-name {
      trigger type shaping-rate (bps | percent percentage);
    }
  }
  classifiers {
    type classifier-name {
      forwarding-class class-name {
        loss-priority (high | low | medium-high | medium-low) code-points [aliases bits ];
      }
      import (classifier-name | default);
    }
  }
}
```

```

    }
  }
  code-point-aliases {
    (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) {
      alias-name bits;
    }
  }
  copy-plp-all;
  drop-profiles {
    profile-name {
      fill-level percentage drop-probability percentage;
      interpolate {
        drop-probability value;
        fill-level value;
      }
    }
  }
  fabric {
    scheduler-map {
      priority (high | low) scheduler scheduler-name;
    }
  }
  forwarding-class-map {
    map-name {
      class class-name queue-num queue-number <restricted-queue queue-number>;
    }
  }
  forwarding-classes {
    class class-name queue-num queue-number priority (high | low);
    queue queue-number class-name priority (high | low);
  }
  forwarding-policy {
    class class-name {
      classification-override {
        forwarding-class class-name;
      }
    }
    next-hop-map map-name {
      forwarding-class class-name {
        discard;
        lsp-next-hop [ lsp-regular-expressions ];
        next-hop [ next-hop-names ];
        non-lsp-next-hop;
      }
    }
  }
  fragmentation-maps {
    map-name {
      forwarding-class class-name {
        drop-timeout milliseconds;
        fragment-threshold bytes;
        multilink-class number;
        no-fragmentation;
      }
    }
  }
}

```

```

host-outbound-traffic {
  dscp-code-point value;
  forwarding-class class-name;
  ieee-802.1 {
    default value;
    rewrite-rules;
  }
  translation-table to-802.1p-from-dscp table-name;
}
interfaces {
  ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
  ...
}
loss-priority-maps {
  frame-relay-de name {
    loss-priority level code-points [alias | bits ];
  }
}
loss-priority-rewrites {
  frame-relay-de name {
    loss-priority level code-point (alias | bits );
  }
}
restricted-queues {
  forwarding-class class-name queue queue-number;
}
rewrite-rules {
  (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | inet-precedence) rewrite-rule {
    forwarding-class class-name {
      loss-priority level code-point (alias | bits );
    }
    import (rewrite-rule | default);
  }
}
routing-instances routing-instance-name {
  classifiers {
    dscp (classifier-name | default);
    dscp-ipv6 (classifier-name | default);
    exp (classifier-name | default);
    ieee-208.1 (classifier-name | default | encapsulated | vlan-tag);
  }
}
scheduler-maps {
  map-name {
    forwarding-class class-name scheduler scheduler-name;
  }
}
schedulers {
  scheduler-name {
    buffer-size (exact | percent percentage | remainder | temporal microseconds);
    drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
      protocol (any | non-tcp | tcp) drop-profile profile-name;
    excess-priority (high | low | medium-high | medium-low);
    excess-rate percent percentage;
    priority (high | low | medium-high | medium-low | strict-high);
  }
}

```

```

        shaping-rate (bps | percent percentage);
        transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
    }
}
traceoptions {
    file <filename> <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
}
traffic-control-profiles {
    profile-name {
        delay-buffer-rate (bps | percent percentage);
        excess-rate (percent percentage | proportion value);
        guaranteed-rate (bps | percent percentage) <burst-size bytes>;
        overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
        scheduler-map map-name;
        shaping-rate (bps | percent percentage) <burst-size bytes>;
    }
}
translation-table {
    to-802.1p-from-dscp table-name {
        to-code-point 3-bit-pattern from-code-points [ 6-bit-patterns ];
    }
    to-dscp-from-dscp table-name {
        to-code-point 6-bit-pattern from-code-points [ 6-bit-patterns ];
    }
    to-dscp-ipv6-from-dscp-ipv6 table-name {
        to-code-point 6-bit-pattern from-code-points [ 6-bit-patterns ];
    }
    to-exp-from-exp table-name {
        to-code-point 3-bit-pattern from-code-points [ 3-bit-patterns ];
    }
    to-inet-precedence-from-inet-precedence table-name {
        to-code-point 3-bit-pattern from-code-points [ 3-bit-patterns ];
    }
}
tri-color;
}

class-of-service {
    interfaces {
        interface-name {
            excess-bandwidth-share (equal | proportional value);
            input-excess-bandwidth-share (equal | proportional value);
            input-scheduler-map map-name;
            input-shaping-rate bps;
            input-traffic-control-profile profile-name;
            input-traffic-control-profile-remaining profile-name;
            output-forwarding-class-map map-name;
            output-traffic-control-profile profile-name;
            output-traffic-control-profile-remaining profile-name;
            scheduler-map map-name;
            scheduler-map-chassis map-name;
            shaping-rate bps;
            unit logical-unit-number {

```

```

adaptive-shaper adaptive-shaper-name;
classifiers {
    dscp (classifier-name | default) {
        family [ inet mpls ];
    }
    dscp-ipv6 (classifier-name | default) {
        family [ inet mpls ];
    }
    exp (classifier-name | default);
    ieee-208.1 (classifier-name | default) <vlan-tag (inner | outer)>;
    ieee-208.1ad (classifier-name | default);
    inet-precedence (classifier-name | default);
}
forwarding-class class-name;
fragmentation-map map-name;
input-scheduler-map map-name;
input-shaping-rate bps;
input-traffic-control-profile profile-name shared-instance instance-name;
loss-priority-maps {
    (map-name | default);
}
loss-priority-rewrites {
    (map-name | default);
}
output-forwarding-class-map map-name;
output-traffic-control-profile profile-name shared-instance instance-name;
per-session-scheduler;
rewrite-rules {
    dscp (rule-name | default) <protocol mpls>;
    dscp-ipv6 (rule-name | default);
    exp (rule-name | default) <protocol [ mpls-any | mpls-inet-both |
        mpls-inet-both-non-vpn ]>;
    exp-push-push-push default;
    exp-swap-push-push default;
    frame-relay-de (rewrite-name | default);
    ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
    ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
    inet-precedence (rewrite-name | default) <protocol mpls>;
}
scheduler-map map-name;
shaping-rate bps;
translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
    to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
}
}
interface-set interface-set-name {
    excess-bandwidth-share (equal | proportional value);
    input-excess-bandwidth-share (equal | proportional value);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
}
}
}


```

**Related Documentation** • *Notational Conventions Used in Junos OS Configuration Hierarchies*

---

## buffer-size (Schedulers)

---

<b>Syntax</b>	buffer-size (percent <i>percentage</i>   remainder   temporal <i>microseconds</i> );
<b>Hierarchy Level</b>	[edit class-of-service schedulers <i>scheduler-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	Specify buffer size.  <div> <b>NOTE:</b> On PTX Series Packet Transport Routers, buffer-size cannot be configured on rate-limited queues.</div>
<b>Default</b>	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.
<b>Options</b>	<b>percent <i>percentage</i></b> —Buffer size as a percentage of the total buffer. <b>Range:</b> 0 through 100  <b>remainder</b> —Remaining buffer available.  <b>temporal <i>microseconds</i></b> —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. <b>Range:</b> The ranges vary by platform as follows: <ul style="list-style-type: none"><li>• For SRX Series Services Gateways: 1 through 2,000,000 microseconds.</li><li>• For vSRX instances: 1 through 32,000,000 microseconds.</li></ul>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	• <i>Scheduler Buffer Size Overview</i>

## drop-profile (Schedulers)

<b>Syntax</b>	<code>drop-profile <i>profile-name</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> <a href="#">drop-profile-map</a> <a href="#">loss-priority</a> (any   low   medium-low   medium-high   high) <a href="#">protocol</a> (any   non-tcp   tcp)]
<b>Release Information</b>	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.
<b>Description</b>	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.
<b>Options</b>	<i>profile-name</i> —Name of the drop profile.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199</a></li> <li>• <a href="#">Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 200</a></li> </ul>

## drop-profile-map (Schedulers)

<b>Syntax</b>	<code>drop-profile-map <a href="#">loss-priority</a> (any   low   medium-low   medium-high   high) <a href="#">protocol</a> (any   non-tcp   tcp) <a href="#">drop-profile (Schedulers)</a> <i>profile-name</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	Define the loss-priority value for a drop profile.  The statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Default Schedulers Overview on page 145</a></li> <li>• <a href="#">Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199</a></li> </ul>

## excess-priority

---

<b>Syntax</b>	<code>excess-priority [ low   medium-low   medium-high   high   none ];</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i>]</code>
<b>Release Information</b>	Statement introduced in Junos OS Release 9.3. Option <b>none</b> introduced in Junos OS Release 11.4.
<b>Description</b>	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.

<b>Options</b>	<p><b>low</b>—Excess traffic for this scheduler has low priority.</p> <p><b>medium-low</b>—Excess traffic for this scheduler has medium-low priority.</p> <p><b>medium-high</b>—Excess traffic for this scheduler has medium-high priority.</p> <p><b>high</b>—Excess traffic for this scheduler has high priority.</p> <p><b>none</b>—System does not demote the priority of guaranteed traffic when the bandwidth exceeds the shaping rate or the guaranteed rate.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Configuring Excess Bandwidth Sharing on IQE PICs</i></li><li>• <a href="#">Bandwidth Sharing on Nonqueueing Packet Forwarding Engines Overview on page 179</a></li><li>• <a href="#">Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 298</a></li></ul>

## excess-rate

<b>Syntax</b>	<code>excess-rate (percent <i>percentage</i>   proportion <i>value</i>);</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ], [edit class-of-service <a href="#">traffic-control-profiles</a> <i>traffic-control-profile-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	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.

<b>Options</b>	<p><b><i>percentage</i></b>—Percentage of the excess bandwidth to share.  <b>Range:</b> 0 through 100 percent  <b>Default:</b> Excess bandwidth is shared in proportion to the configured transmit rate of each queue.</p> <p><b><i>value</i></b>—(M Series, MX Series, T Series routers and EX Series switches only) Proportion of the excess bandwidth to share. Option available at the [edit class-of-service <b>traffic-class-profiles</b> <i>traffic-control-profile-name</i>] hierarchy level only.  <b>Range:</b> 0 through 1000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Scheduler Transmission Rate on page 172</a></li> <li>• <a href="#">Configuring Excess Bandwidth Sharing on IQE PICs</a></li> <li>• <a href="#">Allocating Excess Bandwidth Among Frame Relay DLCIs on Multiservices PICs</a></li> <li>• <a href="#">Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 298</a></li> </ul>

## forwarding-class (Interfaces)

---

<b>Syntax</b>	<code>forwarding-class <i>class-name</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 12.2 for ACX Series routers.
<b>Description</b>	Associate a forwarding class configuration or default mapping with a specific interface.
<b>Options</b>	<i>class-name</i> —Name of the forwarding class.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Applying Forwarding Classes to Interfaces on page 134</a></li></ul>

## interfaces

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

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

**Hierarchy Level** [edit class-of-service]

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

**Description** Configure interface-specific CoS properties for incoming packets.

**Options** The remaining statements are explained separately.

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

**Related Documentation**

- *Overview of BA Classifier Types*
- [Configuring Rewrite Rules on page 215](#)

## loss-priority (Scheduler Drop Profiles)

<b>Syntax</b>	loss-priority (any   high   low   medium-high   medium-low);
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> <a href="#">drop-profile-map</a> ]
<b>Release Information</b>	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.
<b>Description</b>	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.
<b>Options</b>	<b>any</b> —The drop profile applies to packets with any PLP.



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

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

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

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


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

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

<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Default Schedulers Overview on page 145</a></li> <li>• <a href="#">Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199</a></li> <li>• <a href="#">Configuring Schedulers for Priority Scheduling on page 195</a></li> <li>• <a href="#">Configuring Tricolor Marking on page 92</a></li> <li>• <a href="#">protocol (Schedulers) on page 359</a></li> </ul>
------------------------------	---

## priority (Schedulers)

---

<b>Syntax</b>	<code>priority <i>priority-level</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	Specify the packet-scheduling priority value.
<b>Options</b>	<p><i>priority-level</i> can be one of the following:</p> <ul style="list-style-type: none"><li>• <b>low</b>—Scheduler has low priority.</li><li>• <b>medium-low</b>—Scheduler has medium-low priority.</li><li>• <b>medium-high</b>—Scheduler has medium-high priority.</li><li>• <b>high</b>—Scheduler has high priority. Assigning high priority to a queue prevents the queue from being underserved.</li><li>• <b>strict-high</b>—Scheduler has strictly high priority. Configure a <b>high</b> priority queue with unlimited transmission bandwidth available to it. As long as it has traffic to send, the <b>strict-high</b> priority queue receives precedence over <b>low</b>, <b>medium-low</b>, and <b>medium-high</b> priority queues, but not <b>high</b> priority queues. You can configure <b>strict-high</b> priority on only one queue per interface.</li></ul>
<div> <b>NOTE:</b> The <b>strict-high</b> 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.</div>	
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Schedulers for Priority Scheduling on page 195</a></li></ul>

## protocol (Schedulers)

<b>Syntax</b>	protocol (any   non-tcp   tcp);
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> <a href="#">drop-profile-map</a> ]
<b>Release Information</b>	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.
<b>Description</b>	Specify the protocol type for the specified scheduler.
<b>Options</b>	<b>any</b> —Accept any protocol type.  <b>non-tcp</b> —(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.

<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Schedulers on page 146</a></li> </ul>

## scheduler-map (Interfaces and Traffic-Control Profiles)

---

<b>Syntax</b>	<code>scheduler-map <i>map-name</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> ], [edit class-of-service interfaces <i>interface-name</i> <i>unit</i> <i>logical-unit-number</i> ], [edit class-of-service <a href="#">traffic-control-profiles</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>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.</p> <p>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.</p>
<b>Options</b>	<i>map-name</i> —Name of the scheduler map.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Schedulers on page 146</a></li><li>• <a href="#">Oversubscribing Interface Bandwidth on page 163</a></li><li>• <a href="#">output-traffic-control-profile on page 544</a></li></ul>

## scheduler-maps (For Most Interface Types)

---

<b>Syntax</b>	<pre>scheduler-maps {   <i>map-name</i> {     <a href="#">forwarding-class</a> <i>class-name</i> <a href="#">scheduler</a> <i>scheduler-name</i>;   } }</pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Specify a scheduler map name and associate it with the scheduler configuration and forwarding class.
<b>Options</b>	<i>map-name</i> —Name of the scheduler map.  The remaining statements are explained separately.  See “ <a href="#">Configuring Schedulers</a> ” on <a href="#">page 146</a> .
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.

## schedulers (CoS)

<b>Syntax</b>	<pre> schedulers {   scheduler-name {     adjust-minimum <i>rate</i>;     adjust-percent <i>percentage</i>;     buffer-size (<i>seconds</i>   percent <i>percentage</i>   remainder   temporal <i>microseconds</i>);     drop-profile-map <i>loss-priority</i> (any   low   medium-low   medium-high   high) <i>protocol</i>       (any   non-tcp   tcp) <i>drop-profile profile-name</i>;     excess-priority [ low   medium-low   medium-high   high   none];     excess-rate (percent <i>percentage</i>   proportion <i>value</i>);     priority <i>priority-level</i>;     shaping-rate (percent <i>percentage</i>   <i>rate</i>);     transmit-rate (percent <i>percentage</i>   <i>rate</i>   remainder) &lt;exact   rate-limit&gt;;   } } </pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Statement introduced in Junos OS Release 12.1X48 for PTX Series routers.</p>
<b>Description</b>	Specify the scheduler name and parameter values.
<b>Options</b>	<p><b><i>scheduler-name</i></b>—Name of the scheduler to be configured.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Schedulers Overview on page 143</a></li> <li>• <a href="#">Default Schedulers Overview on page 145</a></li> <li>• <a href="#">Configuring Schedulers on page 146</a></li> <li>• <a href="#">Configuring a Scheduler</a></li> </ul>

## transmit-rate (Schedulers)

<b>Syntax</b>	<code>transmit-rate (rate   percent <i>percentage</i>   remainder) &lt;exact   rate-limit&gt;;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ]
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p><b>rate-limit</b> option introduced in Junos OS Release 8.3. Applied to the Multiservices PICs in Junos OS Release 9.4.</p> <p>Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers.</p> <p>Statement introduced in Junos OS Release 12.2 for ACX Series Routers.</p>
<b>Description</b>	Specify the transmit rate or percentage for a scheduler.
<b>Default</b>	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.
<b>Options</b>	<p><b>exact</b>—(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 Packet Transport Routers, this option is allowed only on the non-strict-high (high, medium-high, medium-low, or low) queues.</p> <p><b>percent <i>percentage</i></b>—Percentage of transmission capacity. A percentage of zero drops all packets in the queue.</p> <p><b>Range:</b> 0 through 100 percent for M, MX and T Series routers and EX Series switches; 1 through 100 percent for PTX Series Packet Transport Routers; 0 through 200 percent for the SONET/SDH OC48/STM16 IQE PIC</p>



### 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 `schedulers` *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.

**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 Packet Transport 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 146](#)
- [Configuring Scheduler Transmission Rate on page 172](#)
- [Understanding Scheduling on PTX Series Routers](#)

## unit

<b>Syntax</b>	<pre> unit logical-unit-number {     classifiers {         type (classifier-name   default) family (mpls   all);     }     forwarding-class class-name;     fragmentation-map map-name;     input-traffic-control-profile profiler-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; } </pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">interfaces</a> interface-name]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.
<b>Options</b>	<p><b>logical-unit-number</b>—Number of the logical unit.</p> <p><b>Range:</b> 0 through 16,384</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>Overview of BA Classifier Types</li> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## CHAPTER 21

# Configuration Statements: BA Classifiers

- [\[edit class-of-service\] Hierarchy Level](#) on page 365
- [classifiers \(Logical Interface\)](#) on page 370
- [classifiers \(Routing Instance\)](#) on page 371
- [classifiers \(Definition\)](#) on page 372
- [classifiers \(Physical Interface\)](#) on page 373
- [code-points](#) on page 374
- [copy-plp-all](#) on page 374
- [dscp \(AS PIC Classifiers\)](#) on page 375
- [dscp \(CoS Interfaces\)](#) on page 375
- [dscp-ipv6 \(CoS Rewrite Rules\)](#) on page 376
- [exp](#) on page 377
- [forwarding-class \(BA Classifiers\)](#) on page 378
- [ieee-802.1 \(Rewrite Rules on Logical Interface\)](#) on page 379
- [ieee-802.1 \(Classifier on Physical Interface\)](#) on page 380
- [ieee-802.1ad](#) on page 381
- [import \(Classifiers\)](#) on page 382
- [inet-precedence \(CoS Rewrite Rules\)](#) on page 382
- [inet-precedence \(Classifier on Physical Interface\)](#) on page 383
- [interfaces](#) on page 384
- [loss-priority \(BA Classifiers\)](#) on page 386
- [routing-instances \(CoS\)](#) on page 387
- [system-defaults](#) on page 388
- [unit](#) on page 389

### [\[edit class-of-service\] Hierarchy Level](#)

---

```
class-of-service {  
  classifiers {  
    type classifier-name {  
      forwarding-class class-name {
```

```

        loss-priority (high | low | medium-high | medium-low) code-points [ aliases bits ];
    }
    import (classifier-name | default);
}
}
code-point-aliases {
    (dscp | dscp-ipv6 | exp | ieee-802.1 | ieee-802.1ad | inet-precedence) {
        alias-name bits;
    }
}
drop-profiles {
    profile-name {
        fill-level percentage drop-probability percentage;
        interpolate {
            drop-probability value;
            fill-level value;
        }
    }
}
fabric {
    scheduler-map {
        priority (high | low) scheduler scheduler-name;
    }
}
forwarding-class-map {
    map-name {
        class class-name queue-num queue-number <restricted-queue queue-number>;
    }
}
forwarding-classes {
    class class-name policing-priority (normal | premium) queue-num queue-number
        priority (high | low);
    queue queue-number class-name policing-priority (normal | premium) priority (high |
        low);
}
forwarding-policy {
    class class-name {
        classification-override {
            forwarding-class class-name;
        }
    }
    next-hop-map map-name {
        forwarding-class class-name {
            discard;
            lsp-next-hop [ lsp-regular-expressions ];
            next-hop [ next-hop-names ];
            non-lsp-next-hop;
        }
    }
}
fragmentation-maps {
    map-name {
        forwarding-class class-name {
            drop-timeout milliseconds;
            fragment-threshold bytes;
            multilink-class number;

```

```

        no-fragmentation;
    }
}
host-outbound-traffic {
    dscp-code-point value;
    forwarding-class class-name;
    ieee-802.1 {
        default value;
        rewrite-rules;
    }
    tcp {
        raise-internet-control-priority;
    }
}
interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
}
restricted-queues {
    forwarding-class class-name queue-number;
}
rewrite-rules {
    (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
    rewrite-rule {
        forwarding-class class-name {
            loss-priority level code-point (alias | bits);
        }
        import (rewrite-rule | default);
    }
}
routing-instances routing-instance-name {
    classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
    }
}
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}
schedulers {
    scheduler-name {
        adjust-minimum value;
        adjust-percent value;
        buffer-size (exact | percent percentage | remainder);
        drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
            protocol any;
        excess-priority (high | low | medium-high | medium-low);
        excess-rate (percent percentage | proportion proportion);
        priority (high | low | medium-high | medium-low | strict-high);
        shaping-rate (bps | percent percentage | burst-size size);
    }
}

```

```

        transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
    }
}
traceoptions {
    file <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
}
traffic-control-profiles {
    profile-name {
        adjust-minimum rate;
        delay-buffer-rate (bps | cps cps | percent percentage);
        excess-rate (percent percentage | proportion value);
        guaranteed-rate (bps | percent percentage) <burst-size bytes>;
        overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
        scheduler-map map-name;
        shaping-rate (bps | percent percentage) <burst-size bytes>;
    }
}
tri-color;
}

class-of-service {
    interfaces {
        interface-name {
            excess-bandwidth-share (equal | proportional value);
            input-excess-bandwidth-share (equal | proportional value);
            input-scheduler-map map-name;
            input-shaping-rate bps;
            input-traffic-control-profile profile-name;
            output-forwarding-class-map map-name;
            output-traffic-control-profile profile-name;
            scheduler-map map-name;
            scheduler-map-chassis (map-name | derived);
            shaping-rate bps;
            unit (logical-unit-number | *) {
                classifiers {
                    dscp (classifier-name | default) {
                        family [ inet mpls ];
                    }
                    dscp-ipv6 (classifier-name | default) {
                        family [ inet mpls ];
                    }
                    exp (classifier-name | default);
                    ieee-208.1 (classifier-name | default) <vlan-tag (inner | outer)>;
                    ieee-208.1ad (classifier-name | default);
                    inet-precedence (classifier-name | default);
                }
                forwarding-class class-name;
                input-scheduler-map map-name;
                input-shaping-rate bps;
                input-traffic-control-profile profile-name shared-instance instance-name;
                loss-priority-maps {
                    (map-name | default);
                }
            }
        }
    }
}

```

```

    loss-priority-rewrites {
        (map-name | default);
    }
    output-forwarding-class-map map-name;
    output-traffic-control-profile profile-name shared-instance instance-name;
    rewrite-rules {
        dscp (rule-name | default) <protocol mpls>;
        dscp-ipv6 (rule-name | default);
        exp (rule-name | default) <protocol [ mpls-any | mpls-inet-both |
            mpls-inet-both-non-vpn ]>;
        exp-push-push-push default;
        exp-swap-push-push default;
        ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        inet-precedence (rewrite-name | default) <protocol mpls>;
    }
    scheduler-map map-name;
    shaping-rate bps;
    translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
        to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
    }
}
interface-set interface-set-name {
    excess-bandwidth-share (equal | proportional value);
    input-excess-bandwidth-share (equal | proportional value);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
}
}
}


```

#### Related Documentation

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## classifiers (Logical Interface)

---

<b>Syntax</b>	<pre>classifiers {     type (classifier-name   default) family (mpls   inet); }</pre>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Apply a CoS aggregate behavior classifier to a logical interface. You can apply a default classifier or one that is previously defined.
<b>Options</b>	<p><b>classifier-name</b>—Name of the aggregate behavior classifier.</p> <p><b>type</b>—Traffic type.</p> <p><b>Values:</b> dscp, dscp-ipv6, exp, ieee-802.1, inet-precedence</p>
<hr/> <div> <b>NOTE:</b> You can only specify a family for the dscp and dscp-ipv6 types.</div> <hr/>	
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Default DSCP and DSCP IPv6 Classifiers on page 42</a></li><li>• <a href="#">Applying Behavior Aggregate Classifiers to Logical Interfaces on page 32</a></li></ul>

## classifiers (Routing Instance)

---

<b>Syntax</b>	<pre> classifiers {   exp (classifier-name   default);   dscp (classifier-name   default);   dscp-ipv6 (classifier-name   default); } </pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">routing-instances</a> <i>routing-instance-name</i> ]
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p><b>dscp</b> and <b>dscp-ipv6</b> support introduced in Junos OS Release 9.6.</p>
<b>Description</b>	For routing instances with VRF table labels enabled, apply a custom Multiprotocol Label Switching (MPLS) EXP classifier or DSCP classifier to the routing instance. You can apply the default classifier or one that is previously defined.
<b>Options</b>	<b>classifier-name</b> —Name of the behavior aggregate MPLS EXP or DSCP classifier.
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Applying MPLS EXP Classifiers to Routing Instances on page 50</a></li> <li>• <a href="#">Applying Behavior Aggregate Classifiers to Logical Interfaces on page 32</a></li> </ul>

## classifiers (Definition)

---

Syntax	<pre>classifiers {     type classifier-name {         import (classifier-name   default);         forwarding-class class-name {             loss-priority level code-points [ aliases ] [ bit-patterns ];         }     } }</pre>
Hierarchy Level	[edit class-of-service], [edit class-of-service <b>routing-instances</b> <i>routing-instance-name</i> ]
Release Information	Statement introduced before Junos OS Release 7.4. <b>ieee-802.1ad</b> option introduced in Junos OS Release 9.2.
Description	Define a CoS aggregate behavior 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.
Options	<b>classifier-name</b> —Name of the aggregate behavior classifier.  <b>type</b> —Traffic type: <b>dscp</b> , <b>dscp-ipv6</b> , <b>exp</b> , <b>ieee-802.1</b> , <b>ieee-802.1ad</b> , <b>inet-precedence</b> .
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"><li>Overview of BA Classifier Types</li><li>Example: Configuring CoS for a PBB Network on MX Series Routers</li></ul>

## classifiers (Physical Interface)

---

<b>Syntax</b>	<pre> classifiers {     type (classifier-name   default) ; } </pre>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access routers.
<b>Description</b>	Apply a CoS aggregate behavior classifier to a physical interface. You can apply a default classifier or one that is previously defined.
<b>Options</b>	<p><b>classifier-name</b>—Name of the aggregate behavior classifier.</p> <p><b>type</b>—Traffic type.</p> <p><b>Values:</b> <b>dscp</b>, <b>ieee-802.1</b>, and <b>inet-precedence</b></p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">dscp on page 375</a></li> <li>• <a href="#">inet-precedence on page 383</a></li> <li>• <a href="#">ieee-802.1 on page 380</a></li> </ul>

## code-points

---

<b>Syntax</b>	<code>code-points ([ <i>aliases</i> ]   [ <i>bit-patterns</i> ] );</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service classifiers type <i>classifier-name</i> forwarding-class <i>class-name</i> loss-priority <i>level</i>]</code>
<b>Release Information</b>	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 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Specify one or more DSCP code-point aliases or bit sets for association with a forwarding class.
<b>Options</b>	<i>aliases</i> —Name of the DSCP alias.  <i>bit-patterns</i> —Value of the code-point bits, in six-bit binary form.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26</a></li><li>• <a href="#">Example: Configuring and Applying a Custom DSCP Behavior Aggregate Classifier</a></li></ul>

## copy-plp-all

---

<b>Syntax</b>	<code>copy-plp-all;</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service]</code>
<b>Release Information</b>	Statement introduced in Junos OS Release 10.3.
<b>Description</b>	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).
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Managing Congestion by Setting Packet Loss Priority for Different Traffic Flows on page 139</a></li></ul>

## dscp (AS PIC Classifiers)

<b>Syntax</b>	<code>dscp (<i>alias</i>   <i>bits</i>);</code>
<b>Hierarchy Level</b>	[edit services cos application-profile <i>profile-name</i> (ftp   sip) (data   video   voice)], [edit services cos rule <i>rule-name</i> term <i>term-name</i> <b>then</b> ], [edit services cos rule <i>rule-name</i> term <i>term-name</i> then (reflexive   reverse)]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.1.
<b>Description</b>	Define the Differentiated Services code point (DSCP) mapping that is applied to the packets.
<b>Options</b>	<b><i>alias</i></b> —Name assigned to a set of CoS markers.  <b><i>bits</i></b> —Mapping value in the packet header.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Configuring Actions in a CoS Rule</i></li> </ul>

## dscp (CoS Interfaces)

<b>Syntax</b>	<code>dscp (<i>classifier-name</i>   default);</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> <b>classifiers</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access routers.
<b>Description</b>	For ACX Series Universal Access routers, map the DSCP field of the incoming packet to the forwarding class and packet loss priority based on the specified DSCP classifier.
<b>Options</b>	<b><i>classifier-name</i></b> —Name of the previously defined DSCP behavior aggregate classifier.  <b>default</b> —Default DSCP behavior aggregate classifier.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.

## dscp-ipv6 (CoS Rewrite Rules)

---

<b>Syntax</b>	<code>dscp-ipv6 (rewrite-name   &lt;default&gt;) protocol mpls;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <a href="#">rewrite-rules</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Support for <b>protocol mpls</b> option introduced in Junos OS Release 10.4R2.
<b>Description</b>	<p>For IPv6 traffic, apply a DSCP rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp-ipv6</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"><li>• On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li><li>• On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li></ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules dscp-ipv6</b>] hierarchy level.</p> <p><b>default</b>—Default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">protocol on page 502</a></li><li>• <a href="#">Setting IPv6 DSCP and MPLS EXP Values Independently on page 217</a></li><li>• <a href="#">Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 220</a></li><li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

**exp**

<b>Syntax</b>	<code>exp (rewrite-name   default) protocol protocol-types;</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS Release 12.2. for ACX series
<b>Description</b>	Apply an MPLS experimental (EXP) rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the <code>[edit class-of-service rewrite-rules exp]</code> hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p>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 <b>mpls-inet-both</b> or <b>mpls-inet-both-non-vpn</b> option at the <code>[edit class-of-service interfaces interface interface-name unit logical-unit-number rewrite-rules exp rewrite-rule-name protocol]</code> hierarchy level. The IP precedence rewrite rules explanation does not apply to ACX Series Universal Access routers.</p> <p>On interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series 3D Universal Edge Routers and EX Series switches, we highly recommend that you configure the <b>default</b> option when you configure a behavior aggregate (BA) classifier that does not include a specific rewrite rule for MPLS packets. Doing so ensures that MPLS exp value is rewritten according to the BA classifier rules configured for forwarding or packet loss priority. This does not apply to ACX Series Universal Access routers.</p> <p><b>protocol-types</b>—Specify one or more protocol matching criteria:</p> <ul style="list-style-type: none"> <li>• <b>mpls-any</b>—Apply to MPLS packets, write MPLS header only.</li> <li>• <b>mpls-inet-both</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header.</li> <li>• <b>mpls-inet-both-non-vpn</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header for only non VPN traffic.</li> </ul>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</a></li> <li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li> </ul>

- [rewrite-rules \(Definition\) on page 504](#)

## **forwarding-class (BA Classifiers)**

---

<b>Syntax</b>	<code>forwarding-class <i>class-name</i> {     <a href="#">loss-priority level</a> <a href="#">code-points</a> [ <i>aliases</i> ] [ <i>bit-patterns</i> ]; }</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service classifiers <i>type classifier-name</i>]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Define forwarding class name and option values.
<b>Options</b>	<i>class-name</i> —Name of the forwarding class.  The remaining statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Behavior Aggregate Classifiers on page 36</a></li></ul>

## ieee-802.1 (Rewrite Rules on Logical Interface)

<b>Syntax</b>	ieee-802.1 ( <i>rewrite-name</i>   default) <b>vlan-tag</b> (outer   outer-and-inner);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. <b>vlan-tag</b> statement introduced in Junos OS Release 8.1.
<b>Description</b>	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.
<b>Options</b>	<b>rewrite-name</b> —Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service rewrite-rules ieee-802.1] hierarchy level.  <b>default</b> —The default mapping.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li> <li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li> <li>• <a href="#">exp on page 377</a></li> <li>• <a href="#">exp-push-push-push on page 487</a></li> <li>• <a href="#">exp-swap-push-push on page 488</a></li> <li>• <a href="#">ieee-802.1ad on page 381</a></li> <li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## ieee-802.1 (Classifier on Physical Interface)

---

<b>Syntax</b>	<code>ieee-802.1 (classifier-name   default) vlan-tag (inner   outer );</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> <b>classifiers</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access routers.
<b>Description</b>	For ACX Series Universal Access 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.
<b>Options</b>	<p><b>vlan-tag inner</b>—In the case of double tagged packets, classify based on the 802.1p of the inner VLAN tag.</p> <p><b>vlan-tag outer</b>—Classify based on the 802.1p of the outermost VLAN tag.</p> <p><b>classifier-name</b>—Name of the previously defined ieee-802.1p behavior aggregate classifier.</p> <p><b>default</b>—Default ieee-802.1p behavior aggregate classifier.</p>
<b>Required Privilege Level</b>	<p><b>interface</b>—To view this statement in the configuration.</p> <p><b>interface-control</b>—To add this statement to the configuration.</p>

## ieee-802.1ad

---

<b>Syntax</b>	ieee-802.1ad ( <i>rewrite-name</i>   default) <b>vlan-tag</b> (outer   outer-and-inner);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 9.2.
<b>Description</b>	Apply a IEEE-802.1ad rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules</b> <b>ieee-802.1ad</b>] hierarchy level.</p> <p><b>default</b>—The default rewrite bit mapping.</p> <p><b>vlan-tag</b>—The rewrite rule is applied to the <b>outer</b> or <b>outer-and-inner</b> VLAN tag.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li> <li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li> <li>• <a href="#">exp on page 377</a></li> <li>• <a href="#">exp-push-push-push on page 487</a></li> <li>• <a href="#">exp-swap-push-push on page 488</a></li> <li>• <a href="#">ieee-802.1 (Rewrite Rules on Logical Interface) on page 379</a></li> <li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## import (Classifiers)

---

<b>Syntax</b>	<code>import (classifier-name   default);</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service classifiers type classifier-name]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Specify a default or previously defined classifier.
<b>Options</b>	<b>classifier-name</b> —Name of the classifier mapping configured at the <code>[edit class-of-service classifiers]</code> hierarchy level.  <b>default</b> —The default classifier mapping.
<b>Required Privilege Level</b>	<b>interface</b> —To view this statement in the configuration. <b>interface-control</b> —To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Overview of BA Classifier Types</i></li></ul>

## inet-precedence (CoS Rewrite Rules)

---

<b>Syntax</b>	<code>inet-precedence (rewrite-name   default);</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Apply a IPv4 precedence rewrite rule.
<b>Options</b>	<b>rewrite-name</b> —Name of a <code>rewrite-rules</code> mapping configured at the <code>[edit class-of-service rewrite-rules inet-precedence]</code> hierarchy level.  <b>default</b> —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.
<b>Required Privilege Level</b>	<b>interface</b> —To view this statement in the configuration. <b>interface-control</b> —To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <i>Applying Rewrite Rules to Output Logical Interfaces</i></li><li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

---

## inet-precedence (Classifier on Physical Interface)

---

<b>Syntax</b>	<code>inet-precedence (<i>classifier-name</i>   default);</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> <b>classifiers</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access routers.
<b>Description</b>	On ACX Series Universal Access 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.
<b>Options</b>	<b><i>classifier-name</i></b> —Name of the previously defined inet-precedence behavior aggregate classifier.  <b>default</b> —Default inet-precedence behavior aggregate classifier.
<b>Required Privilege Level</b>	<b>interface</b> —To view this statement in the configuration. <b>interface-control</b> —To add this statement to the configuration.

## interfaces

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

```


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

```

<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Interface-set level added in Junos OS Release 8.5.
<b>Description</b>	Configure interface-specific CoS properties for incoming packets.
<b>Options</b>	The remaining statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Overview of BA Classifier Types</i></li> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## loss-priority (BA Classifiers)

---

<b>Syntax</b>	<code>loss-priority <i>level</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service classifiers <i>type classifier-name</i> forwarding-class <i>class-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Specify packet loss priority value for a specific set of code-point aliases and bit patterns.
<b>Options</b>	<p><i>level</i> can be one of the following:</p> <ul style="list-style-type: none"><li>• <b>high</b>—Packet has high loss priority.</li><li>• <b>medium-high</b>—Packet has medium-high loss priority.</li><li>• <b>medium-low</b>—Packet has medium-low loss priority.</li><li>• <b>low</b>—Packet has low loss priority.</li></ul>
<div> <b>NOTE:</b> <b>medium-low</b> priority is not supported on PTX1000 routers.</div>	
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26</a></li><li>• <a href="#">Configuring Tricolor Marking on page 92</a></li></ul>

## routing-instances (CoS)

<b>Syntax</b>	<pre> routing-instances <i>routing-instance-name</i> {   classifiers {     dscp (<i>classifier-name</i>   default);     dscp-ipv6 (<i>classifier-name</i>   default);     exp (<i>classifier-name</i>   default);   } } </pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	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 MPLS EXP classifier or one that is previously defined.
<b>Default</b>	If you do not include this statement, the default MPLS EXP classifier is applied to the routing instance. When no DSCP classifier is configured, the default MPLS EXP classifier is applied.
<b>Options</b>	<p><i>routing-instance-name</i>—Name of a routing instance.</p> <p><i>classifier-name</i>—Name of the behavior aggregate MPLS EXP classifier or DSCP classifier.</p>
<b>Required Privilege Level</b>	<p>routing—To view this statement in the configuration.</p> <p>routing-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Forwarding Classes on page 116</a></li> <li>• <i>Applying Custom MPLS EXP Classifiers to Routing Instances in Layer 3 VPNs</i></li> </ul>

## 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.

## unit

<b>Syntax</b>	<pre> unit <i>logical-unit-number</i> {   classifiers {     type (<i>classifier-name</i>   default) family (mpls   all);   }   forwarding-class <i>class-name</i>;   fragmentation-map <i>map-name</i>;   input-traffic-control-profile <i>profiler-name</i> shared-instance <i>instance-name</i>;   output-traffic-control-profile <i>profile-name</i> shared-instance <i>instance-name</i>;   per-session-scheduler;   rewrite-rules {     dscp (<i>rewrite-name</i>   default);     dscp-ipv6 (<i>rewrite-name</i>   default);     exp (<i>rewrite-name</i>   default) <i>protocol</i> <i>protocol-types</i>;     exp-push-push-push default;     exp-swap-push-push default;     ieee-802.1 (<i>rewrite-name</i>   default) <i>vlan-tag</i> (outer   outer-and-inner);     inet-precedence (<i>rewrite-name</i>   default);   }   scheduler-map <i>map-name</i>;   shaping-rate <i>rate</i>; } </pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">interfaces</a> <i>interface-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.
<b>Options</b>	<p><b><i>logical-unit-number</i></b>—Number of the logical unit.</p> <p><b>Range:</b> 0 through 16,384</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>Overview of BA Classifier Types</li> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>



# Configuration Statements: Code Point Aliases

- [\[edit class-of-service\] Hierarchy Level on page 391](#)
- [code-point-aliases on page 395](#)
- [dscp \(Rewrite Rules\) on page 396](#)
- [dscp-ipv6 \(CoS Rewrite Rules\) on page 397](#)
- [exp on page 398](#)
- [ieee-802.1 \(Rewrite Rules on Logical Interface\) on page 399](#)
- [inet-precedence \(CoS Rewrite Rules\) on page 400](#)

## [\[edit class-of-service\] Hierarchy Level](#)

---

```

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

```

```

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

```

```

}
rewrite-rules {
  (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
  rewrite-rule {
    forwarding-class class-name {
      loss-priority level code-point (alias | bits);
    }
    import (rewrite-rule | default);
  }
}
routing-instances routing-instance-name {
  classifiers {
    dscp (classifier-name | default);
    dscp-ipv6 (classifier-name | default);
    exp (classifier-name | default);
    ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
  }
}
scheduler-maps {
  map-name {
    forwarding-class class-name scheduler scheduler-name;
  }
}
schedulers {
  scheduler-name {
    adjust-minimum value;
    adjust-percent value;
    buffer-size (exact | percent percentage | remainder);
    drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
      protocol any;
    excess-priority (high | low | medium-high | medium-low);
    excess-rate (percent percentage | proportion proportion);
    priority (high | low | medium-high | medium-low | strict-high);
    shaping-rate (bps | percent percentage | burst-size size);
    transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
  }
}
traceoptions {
  file <files number> <match regular-expression> <size maximum-file-size>
    <world-readable | no-world-readable>;
  flag flag;
  no-remote-trace;
}
traffic-control-profiles {
  profile-name {
    adjust-minimum rate;
    delay-buffer-rate (bps | cps cps | percent percentage);
    excess-rate (percent percentage | proportion value);
    guaranteed-rate (bps | percent percentage) <burst-size bytes>;
    overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
    scheduler-map map-name;
    shaping-rate (bps | percent percentage) <burst-size bytes>;
  }
}
tri-color;
}

```

```

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

```

```

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

```

**Related Documentation** • [Notational Conventions Used in Junos OS Configuration Hierarchies](#)

## code-point-aliases

<b>Syntax</b>	<pre> code-point-aliases {     type {         alias-name <i>bits</i>;     } } </pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Define an alias for a CoS marker.
<b>Options</b>	<p><b><i>alias-name</i></b>—Name of the code-point alias.</p> <p><b><i>bits</i></b>—6-bit value of the code-point bits, in decimal form.</p> <p><b><i>type</i></b>—CoS marker type.</p> <p><b>Values:</b> dscp, dscp-ipv6, exp, ieee-802.1, ieee-802.1ad, inet-precedence</p>
<b>Usage Guidelines</b>	See <a href="#">“Defining Aliases for CoS Value Bit Patterns” on page 30</a> .
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	• <a href="#">Defining Aliases for CoS Value Bit Patterns on page 30</a>

## dscp (Rewrite Rules)

---

<b>Syntax</b>	<code>dscp</code> ( <i>rewrite-name</i>   <code>default</code> ) <code>protocol mpls</code> ;
<b>Hierarchy Level</b>	[ <code>edit class-of-service interfaces</code> <i>interface-name</i> <code>unit</code> <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>For IPv4 traffic, apply a Differentiated Services (DiffServ) code point (DSCP) rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"><li>• On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li><li>• On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li></ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [<code>edit class-of-service rewrite-rules dscp</code>] hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p><code>interface</code>—To view this statement in the configuration.</p> <p><code>interface-control</code>—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li><li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li><li>• <a href="#">Rewriting MPLS and IPv4 Packet Headers</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## dscp-ipv6 (CoS Rewrite Rules)

<b>Syntax</b>	<code>dscp-ipv6 (rewrite-name   &lt;default&gt;) protocol mpls;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Support for <b>protocol mpls</b> option introduced in Junos OS Release 10.4R2.
<b>Description</b>	<p>For IPv6 traffic, apply a DSCP rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp-ipv6</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"> <li>On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li> <li>On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li> </ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules dscp-ipv6</b>] hierarchy level.</p> <p><b>default</b>—Default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">protocol on page 502</a></li> <li>• <a href="#">Setting IPv6 DSCP and MPLS EXP Values Independently on page 217</a></li> <li>• <a href="#">Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 220</a></li> <li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## exp

<b>Syntax</b>	<code>exp (rewrite-name   default) protocol protocol-types;</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS Release 12.2. for ACX series
<b>Description</b>	Apply an MPLS experimental (EXP) rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the <code>[edit class-of-service rewrite-rules exp]</code> hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p>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 <b>mpls-inet-both</b> or <b>mpls-inet-both-non-vpn</b> option at the <code>[edit class-of-service interfaces interface interface-name unit logical-unit-number rewrite-rules exp rewrite-rule-name protocol]</code> hierarchy level. The IP precedence rewrite rules explanation does not apply to ACX Series Universal Access routers.</p> <p>On interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series 3D Universal Edge Routers and EX Series switches, we highly recommend that you configure the <b>default</b> option when you configure a behavior aggregate (BA) classifier that does not include a specific rewrite rule for MPLS packets. Doing so ensures that MPLS exp value is rewritten according to the BA classifier rules configured for forwarding or packet loss priority. This does not apply to ACX Series Universal Access routers.</p> <p><b>protocol-types</b>—Specify one or more protocol matching criteria:</p> <ul style="list-style-type: none"> <li>• <b>mpls-any</b>—Apply to MPLS packets, write MPLS header only.</li> <li>• <b>mpls-inet-both</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header.</li> <li>• <b>mpls-inet-both-non-vpn</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header for only non VPN traffic.</li> </ul>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</a></li> <li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li> </ul>

- [rewrite-rules \(Definition\) on page 504](#)

## ieee-802.1 (Rewrite Rules on Logical Interface)

<b>Syntax</b>	ieee-802.1 ( <i>rewrite-name</i>   default) <b>vlan-tag</b> (outer   outer-and-inner);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. <b>vlan-tag</b> statement introduced in Junos OS Release 8.1.
<b>Description</b>	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.
<b>Options</b>	<b>rewrite-name</b> —Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules</b> <b>ieee-802.1</b> ] hierarchy level.  <b>default</b> —The default mapping.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li> <li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li> <li>• <a href="#">exp on page 377</a></li> <li>• <a href="#">exp-push-push-push on page 487</a></li> <li>• <a href="#">exp-swap-push-push on page 488</a></li> <li>• <a href="#">ieee-802.1ad on page 381</a></li> <li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## inet-precedence (CoS Rewrite Rules)

---

<b>Syntax</b>	<code>inet-precedence (<i>rewrite-name</i>   default);</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <a href="#">rewrite-rules</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Apply a IPv4 precedence rewrite rule.
<b>Options</b>	<p><b><i>rewrite-name</i></b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service rewrite-rules <b>inet-precedence</b>] hierarchy level.</p> <p><b>default</b>—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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li><li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## CHAPTER 23

# Configuration Statements: Forwarding Policy Options

- [\[edit class-of-service\] Hierarchy Level](#) on page 401
- [class \(Forwarding Classes\)](#) on page 406
- [forwarding-class \(Interfaces\)](#) on page 406
- [forwarding-class \(Restricted Queues\)](#) on page 407
- [forwarding-classes](#) on page 408
- [forwarding-classes-interface-specific](#) on page 409
- [interfaces](#) on page 410
- [output-forwarding-class-map](#) on page 411
- [priority \(Fabric Priority\)](#) on page 412
- [queue \(Global Queues\)](#) on page 413
- [queue \(Restricted Queues\)](#) on page 413
- [restricted-queues](#) on page 414
- [unit](#) on page 415

### [\[edit class-of-service\] Hierarchy Level](#)

---

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

```
        interpolate {
            drop-probability value;
            fill-level value;
        }
    }
}
fabric {
    scheduler-map {
        priority (high | low) scheduler scheduler-name;
    }
}
forwarding-class-map {
    map-name {
        class class-name queue-num queue-number <restricted-queue queue-number>;
    }
}
forwarding-classes {
    class class-name policing-priority (normal | premium) queue-num queue-number
        priority (high | low);
    queue queue-number class-name policing-priority (normal | premium) priority (high |
        low);
}
forwarding-policy {
    class class-name {
        classification-override {
            forwarding-class class-name;
        }
    }
}
next-hop-map map-name {
    forwarding-class class-name {
        discard;
        lsp-next-hop [ lsp-regular-expressions ];
        next-hop [ next-hop-names ];
        non-lsp-next-hop;
    }
}
}
fragmentation-maps {
    map-name {
        forwarding-class class-name {
            drop-timeout milliseconds;
            fragment-threshold bytes;
            multilink-class number;
            no-fragmentation;
        }
    }
}
}
host-outbound-traffic {
    dscp-code-point value;
    forwarding-class class-name;
    ieee-802.1 {
        default value;
        rewrite-rules;
    }
}
tcp {
    raise-internet-control-priority;
```

```

    }
  }
  interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
  }
}
restricted-queues {
  forwarding-class class-name queue-number;
}
rewrite-rules {
  (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
  rewrite-rule {
    forwarding-class class-name {
      loss-priority level code-point (alias | bits);
    }
    import (rewrite-rule | default);
  }
}
routing-instances routing-instance-name {
  classifiers {
    dscp (classifier-name | default);
    dscp-ipv6 (classifier-name | default);
    exp (classifier-name | default);
    ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
  }
}
scheduler-maps {
  map-name {
    forwarding-class class-name scheduler scheduler-name;
  }
}
schedulers {
  scheduler-name {
    adjust-minimum value;
    adjust-percent value;
    buffer-size (exact | percent percentage | remainder);
    drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
      protocol any;
    excess-priority (high | low | medium-high | medium-low);
    excess-rate (percent percentage | proportion proportion);
    priority (high | low | medium-high | medium-low | strict-high);
    shaping-rate (bps | percent percentage | burst-size size);
    transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
  }
}
traceoptions {
  file <files number> <match regular-expression> <size maximum-file-size>
    <world-readable | no-world-readable>;
  flag flag;
  no-remote-trace;
}
traffic-control-profiles {
  profile-name {
    adjust-minimum rate;
    delay-buffer-rate (bps | cps cps | percent percentage);
  }
}

```

```
    excess-rate (percent percentage | proportion value);
    guaranteed-rate (bps | percent percentage) <burst-size bytes>;
    overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
    scheduler-map map-name;
    shaping-rate (bps | percent percentage) <burst-size bytes>;
  }
}
tri-color;
}

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

```

        ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        inet-precedence (rewrite-name | default) <protocol mpls>;
    }
    scheduler-map map-name;
    shaping-rate bps;
    translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
        to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
}
}
interface-set interface-set-name {
    excess-bandwidth-share (equal | proportional value);
    input-excess-bandwidth-share (equal | proportional value);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
}
}
}

```

**Related  
Documentation**

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## class (Forwarding Classes)

---

<b>Syntax</b>	<code>class <i>class-name</i> queue-num <i>queue-number</i> <i>priority</i> (high   low);</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">forwarding-classes</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.1.
<b>Description</b>	<p>On M120 , M320, MX Series routers, and T Series routers only, specify the output transmission queue to which to map all input from an associated forwarding class.</p> <p>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 <b>queue</b> statement instead of the <b>class</b> statement at the [edit class-of-service <a href="#">forwarding-classes</a>] hierarchy level.</p>
<b>Options</b>	<p><b><i>class-name</i></b>—Name of forwarding class.</p> <p><b><i>queue-number</i></b>—Output queue number.</p> <p><b>Range:</b> 0 through 15. Some T Series router PICs are restricted to 0 through 3.</p> <p>The remaining statement is explained separately.</p>
<b>Usage Guidelines</b>	See “ <a href="#">Configuring Forwarding Classes</a> ” on page 116.
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">queue (Global Queues) on page 413</a></li></ul>

## forwarding-class (Interfaces)

---

<b>Syntax</b>	<code>forwarding-class <i>class-name</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> ]
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Statement introduced in Junos OS Release 12.2 for ACX Series routers.</p>
<b>Description</b>	Associate a forwarding class configuration or default mapping with a specific interface.
<b>Options</b>	<b><i>class-name</i></b> —Name of the forwarding class.
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Applying Forwarding Classes to Interfaces on page 134</a></li></ul>


## forwarding-class (Restricted Queues)

---

<b>Syntax</b>	forwarding-class <i>class-name</i> <b>queue</b> <i>queue-number</i> ;
<b>Hierarchy Level</b>	[edit class-of-service <b>restricted-queues</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	For M320 and T Series routers only, map forwarding classes to restricted queues. You can map up to eight forwarding classes to restricted queues.
<b>Options</b>	<i>class-name</i> —Name of the forwarding class.  The remaining statement is explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.

## forwarding-classes

---

Syntax	<pre>forwarding-classes {     class queue-num queue-number priority (high   low);     queue queue-number class-name priority (high   low) [ policing-priority (premium   normal) ]; }</pre>
Hierarchy Level	[edit class-of-service]
Release Information	Statement introduced before Junos OS Release 7.4. <b>policing-priority</b> option introduced in Junos OS Release 9.5. Statement introduced on PTX Series Packet Transport Switches in Junos OS Release 12.1.
Description	Associate the forwarding class with a queue name and number. For M320, MX Series, and T Series routers only, you can configure fabric priority queuing by including the <b>priority</b> statement. For Enhanced IQ PICs, you can include the <b>policing-priority</b> option.
<div> <b>NOTE:</b> The priority add policing-priority options are not supported on PTX Series Packet Transport Switches.</div>	
The statements are explained separately.	
Usage Guidelines	See <a href="#">“Configuring Forwarding Classes” on page 116</a> , <i>Overriding Fabric Priority Queuing</i> , and <i>Example: Configuring CoS for a PBB Network on MX Series Routers</i> . For the <b>policing-priority</b> option, see <i>Configuring Layer 2 Policers on IQE PICs</i> . For classification by egress interface, see <i>Classifying Packets by Egress Interface</i> .
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.

## forwarding-classes-interface-specific

<b>Syntax</b>	forwarding-classes-interface-specific <i>forwarding-class-map-name</i> { class <i>class-name</i> queue-num <i>queue-number</i> [ restricted-queue <i>queue-number</i> ]; }
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced in Junos OS Release 9.6.
<b>Description</b>	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.
<b>Options</b>	<p><i>class-name</i>—Name of the forwarding class.</p> <p><i>forwarding-class-map-name</i>—Name of the forwarding class map for traffic.</p> <p><i>queue-number</i>—Number of the egress queue.</p> <p><b>Range:</b> 0 through 3 or 7, depending on chassis and configuration</p>
<b>Usage Guidelines</b>	See <a href="#">“Configuring Forwarding Classes” on page 116</a> and <i>Classifying Packets by Egress Interface</i> .
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">output-forwarding-class-map on page 411</a></li> </ul>

## interfaces

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

```

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

```

Hierarchy Level	[edit class-of-service]
Release Information	Statement introduced before Junos OS Release 7.4. Interface-set level added in Junos OS Release 8.5.
Description	Configure interface-specific CoS properties for incoming packets.
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	<ul style="list-style-type: none"> <li>• <i>Overview of BA Classifier Types</i></li> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## output-forwarding-class-map

Syntax	output-forwarding-class-map <i>forwarding-class-map-name</i> ;
Hierarchy Level	[edit class-of-service <a href="#">forwarding-class-map</a> ]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	Apply a configured forwarding class map to a logical interface.
Options	<i>forwarding-class-map-name</i> —Name of a forwarding class mapping configured at the [edit class-of-service forwarding-classes-interface-specific] hierarchy level.
Usage Guidelines	<i>Classifying Packets by Egress Interface</i>
Required Privilege Level	interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> <li>• <a href="#">forwarding-class-map on page 409</a></li> </ul>

## priority (Fabric Priority)

---

<b>Syntax</b>	priority (high   low);
<b>Hierarchy Level</b>	[edit class-of-service forwarding-classes class class-name queue-num queue-number], [edit class-of-service forwarding-classes queue queue-number class-name]
<b>Release Information</b>	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.
<b>Description</b>	<p>For M320 routers, MX Series routers, and T Series routers only, specify a fabric priority value.</p> <p>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.</p>
<b>Options</b>	<p><b>low</b>—Forwarding class's fabric queuing has low priority.</p> <p><b>high</b>—Forwarding class's fabric queuing has high priority.</p>
<b>Usage Guidelines</b>	See <i>Overriding Fabric Priority Queuing</i> and <i>Configuring Up to 16 Forwarding Classes</i> .
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.

## queue (Global Queues)

<b>Syntax</b>	<code>queue <i>queue-number</i> <i>class-name</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">forwarding-classes</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>Specify the output transmission queue to which to map all input from an associated forwarding class.</p> <p>On M120, M320, MX Series, and T Series routers, 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 <b>class</b> statement instead of the <b>queue</b> statement at the <a href="#">[edit class-of-service forwarding-classes]</a> hierarchy level.</p>
<b>Options</b>	<p><b><i>class-name</i></b>—Name of forwarding class.</p> <p><b><i>queue-number</i></b>—Output queue number.</p> <p><b>Range:</b> For M Series routers, 0 through 3. For M120, M320, MX Series, and T Series routers, 0 through 7. Some T Series router PICs are restricted to 0 through 3.</p>
<b>Usage Guidelines</b>	See “ <a href="#">Configuring Forwarding Classes</a> ” on page 116.
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li><a href="#">class (Forwarding Classes) on page 406</a></li> </ul>

## queue (Restricted Queues)

<b>Syntax</b>	<code>queue <i>queue-number</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">restricted-queues forwarding-class</a> <i>class-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	For M320, MX Series, T Series routers and EX Series switches only, map forwarding classes to restricted queues.
<b>Options</b>	<p><b><i>queue-number</i></b>—Output queue number.</p> <p><b>Range:</b> 0 through 3.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>

## restricted-queues

---

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

**Hierarchy Level**    [edit class-of-service]

**Release Information**    Statement introduced before Junos OS Release 7.4.

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

The remaining statements are explained separately.

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

## unit

<b>Syntax</b>	<pre> unit <i>logical-unit-number</i> {   classifiers {     type (<i>classifier-name</i>   default) family (mpls   all);   }   forwarding-class <i>class-name</i>;   fragmentation-map <i>map-name</i>;   input-traffic-control-profile <i>profiler-name</i> shared-instance <i>instance-name</i>;   output-traffic-control-profile <i>profile-name</i> shared-instance <i>instance-name</i>;   per-session-scheduler;   rewrite-rules {     dscp (<i>rewrite-name</i>   default);     dscp-ipv6 (<i>rewrite-name</i>   default);     exp (<i>rewrite-name</i>   default) <i>protocol</i> <i>protocol-types</i>;     exp-push-push-push default;     exp-swap-push-push default;     ieee-802.1 (<i>rewrite-name</i>   default) <i>vlan-tag</i> (outer   outer-and-inner);     inet-precedence (<i>rewrite-name</i>   default);   }   scheduler-map <i>map-name</i>;   shaping-rate <i>rate</i>; } </pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">interfaces</a> <i>interface-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.
<b>Options</b>	<p><b><i>logical-unit-number</i></b>—Number of the logical unit.</p> <p><b>Range:</b> 0 through 16,384</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>Overview of BA Classifier Types</li> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>



# Configuration Statements: MIC and MPC Interfaces

- [\[edit class-of-service\] Hierarchy Level on page 417](#)
- [adjust-minimum on page 422](#)
- [adjust-percent on page 422](#)
- [excess-priority on page 423](#)
- [excess-rate on page 424](#)
- [excess-rate-high on page 425](#)
- [excess-rate-low on page 425](#)
- [schedulers \(CoS\) on page 426](#)
- [shaping-rate-excess-high on page 427](#)
- [shaping-rate-excess-low on page 428](#)
- [shaping-rate-priority-high on page 429](#)
- [shaping-rate-priority-low on page 430](#)
- [shaping-rate-priority-medium on page 431](#)
- [traffic-control-profiles on page 432](#)

## [\[edit class-of-service\] Hierarchy Level](#)

---

```
class-of-service {
  adaptive-shapers {
    adaptive-shaper-name {
      trigger type shaping-rate (bps | percent percentage);
    }
  }
  classifiers {
    type classifier-name {
      forwarding-class class-name {
        loss-priority (high | low | medium-high | medium-low) code-points [aliases bits ];
      }
      import (classifier-name | default);
    }
  }
  code-point-aliases {
```

```
(dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) {
    alias-name bits;
}
}
copy-plp-all;
drop-profiles {
    profile-name {
        fill-level percentage drop-probability percentage;
        interpolate {
            drop-probability value;
            fill-level value;
        }
    }
}
fabric {
    scheduler-map {
        priority (high | low) scheduler scheduler-name;
    }
}
forwarding-class-map {
    map-name {
        class class-name queue-num queue-number <restricted-queue queue-number>;
    }
}
forwarding-classes {
    class class-name queue-num queue-number priority (high | low);
    queue queue-number class-name priority (high | low);
}
forwarding-policy {
    class class-name {
        classification-override {
            forwarding-class class-name;
        }
    }
    next-hop-map map-name {
        forwarding-class class-name {
            discard;
            lsp-next-hop [ lsp-regular-expressions ];
            next-hop [ next-hop-names ];
            non-lsp-next-hop;
        }
    }
}
fragmentation-maps {
    map-name {
        forwarding-class class-name {
            drop-timeout milliseconds;
            fragment-threshold bytes;
            multilink-class number;
            no-fragmentation;
        }
    }
}
host-outbound-traffic {
    dscp-code-point value;
    forwarding-class class-name;
```

```

ieee-802.1 {
    default value;
    rewrite-rules;
}
translation-table to-802.1p-from-dscp table-name;
}
interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
}
loss-priority-maps {
    frame-relay-de name {
        loss-priority level code-points [alias | bits];
    }
}
loss-priority-rewrites {
    frame-relay-de name {
        loss-priority level code-point (alias | bits);
    }
}
restricted-queues {
    forwarding-class class-name queue queue-number;
}
rewrite-rules {
    (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | inet-precedence) rewrite-rule {
        forwarding-class class-name {
            loss-priority level code-point (alias | bits);
        }
        import (rewrite-rule | default);
    }
}
routing-instances routing-instance-name {
    classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag);
    }
}
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}
schedulers {
    scheduler-name {
        buffer-size (exact | percent percentage | remainder | temporal microseconds);
        drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
            protocol (any | non-tcp | tcp) drop-profile profile-name;
        excess-priority (high | low | medium-high | medium-low);
        excess-rate percent percentage;
        priority (high | low | medium-high | medium-low | strict-high);
        shaping-rate (bps | percent percentage);
        transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
    }
}

```

```

}
traceoptions {
  file <filename> <files number> <match regular-expression> <size maximum-file-size>
    <world-readable | no-world-readable>;
  flag <flag>;
  no-remote-trace;
}
traffic-control-profiles {
  profile-name {
    delay-buffer-rate (bps | percent percentage);
    excess-rate (percent percentage | proportion value);
    guaranteed-rate (bps | percent percentage) <burst-size bytes>;
    overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
    scheduler-map map-name;
    shaping-rate (bps | percent percentage) <burst-size bytes>;
  }
}
translation-table {
  to-802.1p-from-dscp table-name {
    to-code-point 3-bit-pattern from-code-points [ 6-bit-patterns ];
  }
  to-dscp-from-dscp table-name {
    to-code-point 6-bit-pattern from-code-points [ 6-bit-patterns ];
  }
  to-dscp-ipv6-from-dscp-ipv6 table-name {
    to-code-point 6-bit-pattern from-code-points [ 6-bit-patterns ];
  }
  to-exp-from-exp table-name {
    to-code-point 3-bit-pattern from-code-points [ 3-bit-patterns ];
  }
  to-inet-precedence-from-inet-precedence table-name {
    to-code-point 3-bit-pattern from-code-points [ 3-bit-patterns ];
  }
}
tri-color;
}

class-of-service {
  interfaces {
    interface-name {
      excess-bandwidth-share (equal | proportional value);
      input-excess-bandwidth-share (equal | proportional value);
      input-scheduler-map map-name;
      input-shaping-rate bps;
      input-traffic-control-profile profile-name;
      input-traffic-control-profile-remaining profile-name;
      output-forwarding-class-map map-name;
      output-traffic-control-profile profile-name;
      output-traffic-control-profile-remaining profile-name;
      scheduler-map map-name;
      scheduler-map-chassis map-name;
      shaping-rate bps;
      unit logical-unit-number {
        adaptive-shaper adaptive-shaper-name;
        classifiers {
          dscp (classifier-name | default) {

```

```

        family [ inet mpls ];
    }
    dscp-ipv6 (classifier-name | default) {
        family [ inet mpls ];
    }
    exp (classifier-name | default);
    ieee-208.1 (classifier-name | default) <vlan-tag (inner | outer)>;
    ieee-208.1ad (classifier-name | default);
    inet-precedence (classifier-name | default);
}
forwarding-class class-name;
fragmentation-map map-name;
input-scheduler-map map-name;
input-shaping-rate bps;
input-traffic-control-profile profile-name shared-instance instance-name;
loss-priority-maps {
    (map-name | default);
}
loss-priority-rewrites {
    (map-name | default);
}
output-forwarding-class-map map-name;
output-traffic-control-profile profile-name shared-instance instance-name;
per-session-scheduler;
rewrite-rules {
    dscp (rule-name | default) <protocol mpls>;
    dscp-ipv6 (rule-name | default);
    exp (rule-name | default) <protocol [ mpls-any | mpls-inet-both |
        mpls-inet-both-non-vpn ]>;
    exp-push-push-push default;
    exp-swap-push-push default;
    frame-relay-de (rewrite-name | default);
    ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
    ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
    inet-precedence (rewrite-name | default) <protocol mpls>;
}
scheduler-map map-name;
shaping-rate bps;
translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
    to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
}
}
interface-set interface-set-name {
    excess-bandwidth-share (equal | proportional value);
    input-excess-bandwidth-share (equal | proportional value);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
}
}
}

```

- Related Documentation**
- *Notational Conventions Used in Junos OS Configuration Hierarchies*

---

## adjust-minimum

---

<b>Syntax</b>	<code>adjust-minimum <i>rate</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ], [edit class-of-service <a href="#">traffic-control-profiles</a> <i>traffic-control-profile-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 11.4.
<b>Description</b>	<p>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.</p> <p>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.</p>
<b>Options</b>	<b>rate</b> —Minimum shaping rate for a node or a queue, in Mbps
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	• <i>Configuring the Minimum Adjusted Shaping Rate on Scheduler Nodes for Subscribers</i>

---

## adjust-percent

---

<b>Syntax</b>	<code>adjust-percent <i>percentage</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 11.4.
<b>Description</b>	For a MIC or MPC interface, determine the percentage of adjustment for the shaping rate of a queue.
<b>Options</b>	<b>percentage</b> —Percentage of the shaping rate to adjust. <b>Range:</b> 0 through 100 percent
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	• <i>Configuring Shaping-Rate Adjustments on Queues</i>

## excess-priority

<b>Syntax</b>	<code>excess-priority [ low   medium-low   medium-high   high   none ];</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 9.3. Option <b>none</b> introduced in Junos OS Release 11.4.
<b>Description</b>	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.

<b>Options</b>	<p><b>low</b>—Excess traffic for this scheduler has low priority.</p> <p><b>medium-low</b>—Excess traffic for this scheduler has medium-low priority.</p> <p><b>medium-high</b>—Excess traffic for this scheduler has medium-high priority.</p> <p><b>high</b>—Excess traffic for this scheduler has high priority.</p> <p><b>none</b>—System does not demote the priority of guaranteed traffic when the bandwidth exceeds the shaping rate or the guaranteed rate.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Configuring Excess Bandwidth Sharing on IQE PICs</i></li> <li>• <a href="#">Bandwidth Sharing on Nonqueueing Packet Forwarding Engines Overview on page 179</a></li> <li>• <a href="#">Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 298</a></li> </ul>

## excess-rate

---

<b>Syntax</b>	<code>excess-rate (percent <i>percentage</i>   proportion <i>value</i>);</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i>],</code> <code>[edit class-of-service <a href="#">traffic-control-profiles</a> <i>traffic-control-profile-name</i>]</code>
<b>Release Information</b>	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.
<b>Description</b>	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.
<div> <b>NOTE:</b> The <code>proportion</code> option provides a greater range of values over the <code>percent</code> option and hence influences the priorities assigned to the queues.</div>	
<b>Options</b>	<p><b><i>percentage</i></b>—Percentage of the excess bandwidth to share. <b>Range:</b> 0 through 100 percent <b>Default:</b> Excess bandwidth is shared in proportion to the configured transmit rate of each queue.</p> <p><b><i>value</i></b>—(M Series, MX Series, T Series routers and EX Series switches only) Proportion of the excess bandwidth to share. Option available at the <code>[edit class-of-service traffic-class-profiles <i>traffic-control-profile-name</i>]</code> hierarchy level only. <b>Range:</b> 0 through 1000</p>
<b>Required Privilege Level</b>	<code>interface</code> —To view this statement in the configuration. <code>interface-control</code> —To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Scheduler Transmission Rate on page 172</a></li><li>• <a href="#">Configuring Excess Bandwidth Sharing on IQE PICs</a></li><li>• <a href="#">Allocating Excess Bandwidth Among Frame Relay DLCIs on Multiservices PICs</a></li><li>• <a href="#">Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 298</a></li></ul>

## excess-rate-high

---

<b>Syntax</b>	<code>excess-rate-high (percent <i>percentage</i>   proportion <i>value</i>);</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles</a> <i>traffic-control-profile-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 11.4.
<b>Description</b>	For a MIC or MPC interface, determine the percentage of excess bandwidth from high-priority traffic to share.
<b>Options</b>	<p><b><i>percentage</i></b>—Percentage of the excess bandwidth to share.  <b>Range:</b> 0 through 100 percent</p> <p><b><i>proportion</i></b>—Proportion of the excess bandwidth to share.  <b>Range:</b> 0 through 1000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 298</a></li> </ul>

## excess-rate-low

---

<b>Syntax</b>	<code>excess-rate-low (percent <i>percentage</i>   proportion <i>value</i>);</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles</a> <i>traffic-control-profile-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 11.4.
<b>Description</b>	For a MIC or MPC interface, determine the percentage of excess bandwidth from low-priority traffic to share.
<b>Options</b>	<p><b><i>percentage</i></b>—Percentage of the excess bandwidth to share.  <b>Range:</b> 0 through 100 percent</p> <p><b><i>value</i></b>—Proportion of the excess bandwidth to share.  <b>Range:</b> 0 through 1000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 298</a></li> </ul>

## schedulers (CoS)

---

Syntax	<pre>schedulers {   scheduler-name {     adjust-minimum <i>rate</i>;     adjust-percent <i>percentage</i>;     buffer-size (<i>seconds</i>   percent <i>percentage</i>   remainder   temporal <i>microseconds</i>);     drop-profile-map <i>loss-priority</i> (any   low   medium-low   medium-high   high) <i>protocol</i>       (any   non-tcp   tcp) <i>drop-profile profile-name</i>;     excess-priority [ low   medium-low   medium-high   high   none];     excess-rate (percent <i>percentage</i>   proportion <i>value</i>);     priority <i>priority-level</i>;     shaping-rate (percent <i>percentage</i>   <i>rate</i>);     transmit-rate (percent <i>percentage</i>   <i>rate</i>   remainder) &lt;exact   rate-limit&gt;;   } }</pre>
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	<i>scheduler-name</i> —Name of the scheduler to be configured.  The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"><li>• <a href="#">Schedulers Overview on page 143</a></li><li>• <a href="#">Default Schedulers Overview on page 145</a></li><li>• <a href="#">Configuring Schedulers on page 146</a></li><li>• <a href="#">Configuring a Scheduler</a></li></ul>

## shaping-rate-excess-high

<b>Syntax</b>	<code>shaping-rate-excess-high <i>rate</i> [ <i>burst-size bytes</i> ];</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles</a> <i>profile-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 10.1.
<b>Description</b>	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.
<b>Default</b>	If you do not include this statement, the default shaping rate for this priority is determined by the <b>shaping-rate</b> statement in the traffic control profile.
<b>Options</b>	<p><b>rate</b>—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 <b>k</b> (1000), <b>m</b> (1,000,000), or <b>g</b> (1,000,000,000).</p> <p><b>Range:</b> 1000 through 6,400,000,000,000</p> <p><b>Default:</b> None</p> <p><b>burst-size <i>bytes</i></b>—Maximum burst size, in bytes.</p> <p><b>Range:</b> 0 through 1,000,000,000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Per-Priority Shaping on MIC and MPC Interfaces Overview</i></li> <li>• <a href="#">Oversubscribing Interface Bandwidth on page 163</a></li> <li>• <i>Configuring Traffic Control Profiles for Shared Scheduling and Shaping</i></li> <li>• <a href="#">shaping-rate-excess-low on page 428</a></li> <li>• <a href="#">shaping-rate-priority-high on page 429</a></li> <li>• <a href="#">shaping-rate-priority-low on page 430</a></li> <li>• <a href="#">shaping-rate-priority-medium on page 431</a></li> </ul>

## shaping-rate-excess-low

---

<b>Syntax</b>	<code>shaping-rate-excess-low rate [ burst-size bytes ];</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles</a> <i>profile-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 10.1.
<b>Description</b>	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.
<b>Default</b>	If you do not include this statement, the default shaping rate for this priority is determined by the <b>shaping-rate</b> statement in the traffic control profile.
<b>Options</b>	<p><b>rate</b>—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 <b>k</b> (1000), <b>m</b> (1,000,000), or <b>g</b> (1,000,000,000).</p> <p><b>Range:</b> 1000 through 6,400,000,000,000</p> <p><b>Default:</b> None</p> <p><b>burst-size bytes</b>—Maximum burst size, in bytes.</p> <p><b>Range:</b> 0 through 1,000,000,000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Per-Priority Shaping on MIC and MPC Interfaces Overview</i></li><li>• <a href="#">Oversubscribing Interface Bandwidth on page 163</a></li><li>• <i>Configuring Traffic Control Profiles for Shared Scheduling and Shaping</i></li><li>• <a href="#">shaping-rate-excess-high on page 427</a></li><li>• <a href="#">shaping-rate-priority-high on page 429</a></li><li>• <a href="#">shaping-rate-priority-low on page 430</a></li><li>• <a href="#">shaping-rate-priority-medium on page 431</a></li></ul>

## shaping-rate-priority-high

<b>Syntax</b>	<code>shaping-rate-priority-high rate [ burst-size bytes ];</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles profile-name</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 10.1.
<b>Description</b>	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.
<b>Default</b>	If you do not include this statement, the default shaping rate for this priority is determined by the <b>shaping-rate</b> statement in the traffic control profile.
<b>Options</b>	<p><b>rate</b>—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 <b>k</b> (1000), <b>m</b> (1,000,000), or <b>g</b> (1,000,000,000).</p> <p><b>Range:</b> 1000 through 6,400,000,000,000</p> <p><b>Default:</b> None</p> <p><b>burst-size bytes</b>—Maximum burst size, in bytes.</p> <p><b>Range:</b> 0 through 1,000,000,000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Per-Priority Shaping on MIC and MPC Interfaces Overview</i></li> <li>• <a href="#">Oversubscribing Interface Bandwidth on page 163</a></li> <li>• <i>Configuring Traffic Control Profiles for Shared Scheduling and Shaping</i></li> <li>• <a href="#">shaping-rate-excess-high on page 427</a></li> <li>• <a href="#">shaping-rate-excess-low on page 428</a></li> <li>• <a href="#">shaping-rate-priority-low on page 430</a></li> <li>• <a href="#">shaping-rate-priority-medium on page 431</a></li> </ul>

## shaping-rate-priority-low

---

<b>Syntax</b>	<code>shaping-rate-priority-low rate [ burst-size <i>bytes</i> ];</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles</a> <i>profile-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 10.1.
<b>Description</b>	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.
<b>Default</b>	If you do not include this statement, the default shaping rate for this priority is determined by the <b>shaping-rate</b> statement in the traffic control profile.
<b>Options</b>	<p><b>rate</b>—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 <b>k</b> (1000), <b>m</b> (1,000,000), or <b>g</b> (1,000,000,000).</p> <p><b>Range:</b> 1000 through 6,400,000,000,000</p> <p><b>Default:</b> None</p> <p><b>burst-size <i>bytes</i></b>—Maximum burst size, in bytes.</p> <p><b>Range:</b> 0 through 1,000,000,000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Per-Priority Shaping on MIC and MPC Interfaces Overview</i></li><li>• <a href="#">Oversubscribing Interface Bandwidth on page 163</a></li><li>• <i>Configuring Traffic Control Profiles for Shared Scheduling and Shaping</i></li><li>• <a href="#">shaping-rate-excess-high on page 427</a></li><li>• <a href="#">shaping-rate-excess-low on page 428</a></li><li>• <a href="#">shaping-rate-priority-high on page 429</a></li><li>• <a href="#">shaping-rate-priority-medium on page 431</a></li></ul>

## shaping-rate-priority-medium

<b>Syntax</b>	<code>shaping-rate-priority-medium rate [ burst-size bytes ];</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles</a> <i>profile-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 10.1.
<b>Description</b>	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.
<b>Default</b>	If you do not include this statement, the default shaping rate for this priority is determined by the <b>shaping-rate</b> statement in the traffic control profile.
<b>Options</b>	<p><b>rate</b>—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 <b>k</b> (1000), <b>m</b> (1,000,000), or <b>g</b> (1,000,000,000).</p> <p><b>Range:</b> 1000 through 6,400,000,000,000</p> <p><b>Default:</b> None</p> <p><b>burst-size bytes</b>—Maximum burst size, in bytes.</p> <p><b>Range:</b> 0 through 1,000,000,000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Per-Priority Shaping on MIC and MPC Interfaces Overview</i></li> <li>• <a href="#">Oversubscribing Interface Bandwidth on page 163</a></li> <li>• <i>Configuring Traffic Control Profiles for Shared Scheduling and Shaping</i></li> <li>• <a href="#">shaping-rate-excess-high on page 427</a></li> <li>• <a href="#">shaping-rate-excess-low on page 428</a></li> <li>• <a href="#">shaping-rate-priority-high on page 429</a></li> <li>• <a href="#">shaping-rate-priority-low on page 430</a></li> </ul>

## 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.

**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 163](#)
- [Understanding Scheduling on PTX Series Routers](#)
- [output-traffic-control-profile on page 544](#)

# Configuration Statements: Packets Using Multifield Classifiers

- [\[edit class-of-service\] Hierarchy Level on page 433](#)
- [transparent on page 437](#)
- [\[edit firewall\] Hierarchy Level on page 437](#)
- [dscp \(Multifield Classifier\) on page 449](#)
- [family \(Multifield Classifier\) on page 450](#)
- [filter \(Configuring\) on page 451](#)
- [forwarding-class \(Multifield Classifiers\) on page 452](#)
- [from on page 452](#)
- [loss-priority \(Firewall Filter\) on page 453](#)
- [loss-priority \(Simple Firewall Filter\) on page 453](#)
- [term \(Simple Filter\) on page 454](#)
- [then \(Services CoS\) on page 455](#)
- [\[edit interfaces\] Hierarchy Level on page 455](#)
- [filter \(Applying to an Interface\) on page 467](#)
- [simple-filter \(Applying to an Interface\) on page 468](#)

## [\[edit class-of-service\] Hierarchy Level](#)

---

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

```

drop-profiles {
  profile-name {
    fill-level percentage drop-probability percentage;
    interpolate {
      drop-probability value;
      fill-level value;
    }
  }
}
fabric {
  scheduler-map {
    priority (high | low) scheduler scheduler-name;
  }
}
forwarding-class-map {
  map-name {
    class class-name queue-num queue-number <restricted-queue queue-number>;
  }
}
forwarding-classes {
  class class-name policing-priority (normal | premium) queue-num queue-number
  priority (high | low);
  queue queue-number class-name policing-priority (normal | premium) priority (high |
  low);
}
forwarding-policy {
  class class-name {
    classification-override {
      forwarding-class class-name;
    }
  }
  next-hop-map map-name {
    forwarding-class class-name {
      discard;
      lsp-next-hop [ lsp-regular-expressions ];
      next-hop [ next-hop-names ];
      non-lsp-next-hop;
    }
  }
}
fragmentation-maps {
  map-name {
    forwarding-class class-name {
      drop-timeout milliseconds;
      fragment-threshold bytes;
      multilink-class number;
      no-fragmentation;
    }
  }
}
host-outbound-traffic {
  dscp-code-point value;
  forwarding-class class-name;
  ieee-802.1 {
    default value;
    rewrite-rules;
  }
}

```

```

    }
    tcp {
        raise-internet-control-priority;
    }
}
interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
}
}
restricted-queues {
    forwarding-class class-name queue-number;
}
rewrite-rules {
    (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
    rewrite-rule {
        forwarding-class class-name {
            loss-priority level code-point (alias | bits);
        }
        import (rewrite-rule | default);
    }
}
routing-instances routing-instance-name {
    classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
    }
}
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}
schedulers {
    scheduler-name {
        adjust-minimum value;
        adjust-percent value;
        buffer-size (exact | percent percentage | remainder);
        drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
            protocol any;
        excess-priority (high | low | medium-high | medium-low);
        excess-rate (percent percentage | proportion proportion);
        priority (high | low | medium-high | medium-low | strict-high);
        shaping-rate (bps | percent percentage | burst-size size);
        transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
    }
}
traceoptions {
    file <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
}
traffic-control-profiles {

```

```

    profile-name {
        adjust-minimum rate;
        delay-buffer-rate (bps | cps cps | percent percentage);
        excess-rate (percent percentage | proportion value);
        guaranteed-rate (bps | percent percentage) <burst-size bytes>;
        overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
        scheduler-map map-name;
        shaping-rate (bps | percent percentage) <burst-size bytes>;
    }
}
tri-color;
}

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

```

```

exp-push-push-push default;
exp-swap-push-push default;
ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
inet-precedence (rewrite-name | default) <protocol mpls>;
}
scheduler-map map-name;
shaping-rate bps;
translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
}
}
interface-set interface-set-name {
excess-bandwidth-share (equal | proportional value);
input-excess-bandwidth-share (equal | proportional value);
input-traffic-control-profile profile-name;
input-traffic-control-profile-remaining profile-name;
internal-node;
output-traffic-control-profile profile-name;
output-traffic-control-profile-remaining profile-name;
}
}
}

```

**Related  
Documentation**

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## transparent

<b>Syntax</b>	transparent;
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> classifiers ieee802.1 vlan-tag]
<b>Release Information</b>	Statement introduced in Junos OS Release 11.2
<b>Description</b>	Packet classification based on the transparent VLAN tag.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.

## [edit firewall] Hierarchy Level

Several statements in the **[edit firewall]** hierarchy are valid at numerous locations within the hierarchy. .

- [Common Firewall Actions on page 438](#)
- [Common IP Firewall Actions on page 438](#)
- [Common IPv4 and IPv6 Firewall Actions on page 439](#)
- [Common IP Firewall Match Conditions on page 439](#)
- [Common IPv4 Firewall Match Conditions on page 440](#)

- [Common Layer 2 Firewall Match Conditions on page 440](#)
- [Complete \[edit firewall\] Hierarchy on page 442](#)

## Common Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on page 326 instead of the statements being repeated.

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

The common firewall actions are as follows:

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

## Common IP Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on page 326 instead of the statements being repeated.

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

The common IP firewall actions are as follows:

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

## Common IPv4 and IPv6 Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on page 326 instead of the statements being repeated.

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

The common IP version 4 (IPv4) and IP version 6 (IPv6) firewall actions are as follows:

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

## Common IP Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on page 326 instead of the statements being repeated.

- **[edit firewall family inet dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in “[Complete \[edit firewall\] Hierarchy](#)” on page 326)
- **[edit firewall family inet filter *filter-name* term *term-name* from]**
- **[edit firewall family inet6 dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in “[Complete \[edit firewall\] Hierarchy](#)” on page 326)
- **[edit firewall family inet6 filter *filter-name* term *term-name* from]**
- **[edit firewall filter *filter-name* term *term-name* from]**

The common IP firewall match conditions are as follows:

```
address {
    ip-prefix</prefix-length> <except>;
}
destination-address {
    ip-prefix</prefix-length> <except>;
}
destination-class [ class-names ] | destination-class-except [ class-names ];
(destination-port [ port-names ] | destination-port-except [ port-names ]);
destination-prefix-list {
    list-name <except>;
}
(forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
icmp-code [ codes ] | icmp-code-except [ codes ];
icmp-type [ types ] | icmp-type-except [ types ];
```

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

## Common IPv4 Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in [“Complete \[edit firewall\] Hierarchy” on page 326](#) instead of the statements being repeated.

- **[edit firewall family inet dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in [“Complete \[edit firewall\] Hierarchy” on page 326](#))
- **[edit firewall family inet filter *filter-name* term *term-name* from]**
- **[edit firewall filter *filter-name* term *term-name* from]**

The common IPv4 firewall match conditions are as follows:

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

## Common Layer 2 Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in [“Complete \[edit firewall\] Hierarchy” on page 326](#) instead of the statements being repeated.

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

The common Layer 2 firewall match conditions are as follows:

```

destination-mac-address {
    mac-address <except>;
}
(destination-port [ port-names ] | destination-port-except [ port-names ]);
(dscp [ code-point-values ] | dscp-except [ code-point-values ]);
(ether-type [ protocol-types ] | ether-type-except [ protocol-types ]);
(forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
 icmp-code [ codes ] | icmp-code-except [ codes ];
 icmp-type [ types ] | icmp-type-except [ types ];
(interface-group [ group-names ] | interface-group-except [ group-names ]);
ip-address {
    ip-prefix</prefix-length> <except>;
}
ip-destination-address {
    ip-prefix</prefix-length> <except>;
}
(ip-precedence [ precedence-names ] | ip-precedence-except [ precedence-names ]);
(ip-protocol [ protocol-names ] | ip-protocol-except [ protocol-names ]);
ip-source-address ip-prefix</prefix-length>;
(learn-vlan-lp-priority [ priorities ] | learn-vlan-lp-priority [ priorities ]);
(learn-vlan-id [ vlan-ids ] | learn-vlan-id-except [ vlan-ids ]);
(loss-priority [ priorities ] | loss-priority-except [ priorities ]);
(port [ port-names ] | port-except [ port-names ]);
source-mac-address {
    mac-address <except>;
}
(source-port [ port-names ] | source-port-except [ port-names ]);
tcp-flags flag;
(traffic-type [ broadcast known-unicast multicast unknown-unicast ] |
 traffic-type-except [ broadcast known-unicast multicast unknown-unicast ]);
(user-vlan-lp-priority [ priorities ] | user-vlan-lp-priority [ priorities ]);
(user-vlan-id [ vlan-ids ] | user-vlan-id-except [ vlan-ids ]);
(vlan-ether-type [ protocol-types ] | vlan-ether-type-except [ protocol-types ]);

```

## Complete [edit firewall] Hierarchy

```
firewall {
  family (any | ethernet-switching | inet | inet6) {
    ... the family subhierarchies appear after the main [edit firewall] hierarchy ...
  }
  filter filter-name {
    accounting-profile [ profile-names ];
    enhanced-mode;
    interface-shared-with;
    interface-specific;
    physical-interface-policer;
    term term-name {
      filter filter-name;
      from {
        ... statements in Common IP Firewall Match Conditions on page 323 AND
        ... statements in Common IPv4 Firewall Match Conditions on page 324 ...
      }
      then {
        ... statements in Common Firewall Actions on page 321 AND
        ... statements in Common IP Firewall Actions on page 322 AND
        ... statements in Common IPv4 and IPv6 Firewall Actions on page 322 ...
      }
    }
  }
}
hierarchical-policer policer-name {
  aggregate {
    if-exceeding {
      bandwidth-limit bps;
      burst-size-limit bytes;
    }
    then {
      discard;
      forwarding-class class-name;
      loss-priority (high | low | medium-high | medium-low);
    }
  }
  logical-interface-policer;
  physical-interface-policer;
  premium {
    if-exceeding {
      bandwidth-limit bps;
      burst-size-limit bytes;
    }
    then {
      discard;
    }
  }
}
shared-bandwidth-policer;
interface-set interface-set-name {
  interface-name;
}
load-balance-group group-name {
  next-hop-group [ group-names ];
}
```

```

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

firewall {
  family any {
    filter filter-name {
      interface-shared;
      term term-name {
        from {
          (forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
          interface interface-name;
          interface-set set-name;
          (loss-priority [ priorities ] | loss-priority-except [ priorities ]);
          (packet-length [ values ] | packet-length-except [ values ]);
        }
        then {
          ... statements in Common Firewall Actions on page 321 PLUS ...
          (accept | discard);
        }
      }
    }
  }
}

```

```
    }  
  }  
}
```

```
firewall {  
  family ethernet-switching {  
    filter filter-name {  
      interface-specific;  
      term term-name {  
        from {  
          destination-address {  
            ip-prefix</prefix-length>;  
          }  
          destination-mac-address {  
            mac-address;  
          }  
          destination-port [ port-names ];  
          destination-prefix-list {  
            list-name;  
          }  
          dot1q-tag [ tag-values ];  
          dot1q-user-priority [ priority-values ];  
          dscp [ code-point-values ];  
          ether-type [ protocol-names ];  
          fragment-flags flag;  
          icmp-code [ codes ];  
          icmp-type [ types ];  
          interface interface-name;  
          is-fragment;  
          precedence [ precedence-names ];  
          protocol [ protocol-names ];  
          source-address {  
            ip-prefix</prefix-length>;  
          }  
          source-mac-address {  
            mac-address;  
          }  
          source-port [ port-names ];  
          source-prefix-list {  
            list-name;  
          }  
          tcp-established;  
          tcp-flags flag;  
          tcp-initial;  
          vlan [ vlan-names ];  
        }  
        then {  
          (accept | discard);  
          analyzer analyzer-name;  
          count counter-name;  
          forwarding-class class-name;  
          interface interface-name;  
          log;  
        }  
      }  
    }  
  }  
}
```

```

        loss-priority (high | low);
        policer policer-name;
        syslog;
        vlan vlan-name;
    }
}
}
}
}

firewall {
    family inet {
        dialer-filter filter-name {
            accounting-profile [ profile-names ];
            term term-name {
                from {
                    ... statements in Common IP Firewall Match Conditions on page 323 AND
                    statements in Common IPv4 Firewall Match Conditions on page 324 EXCEPT
                    FOR ...
                    (ah-spi [ values ] | ah-spi-except [ values ]); # NOT valid at this level
                    (destination-class [ class-names ] |
                     destination-class-except [ class-names ]); # NOT valid at this level
                    interface interface-name; # NOT valid at this level
                    (loss-priority [ priorities ] | loss-priority-except [ priorities ]); # NOT valid at
                     this level
                    service-filter-hit; # NOT valid at this level
                    (source-class [ class-names ] | source-class-except [ class-names ]); # NOT
                     valid at this level
                }
                then {
                    (ignore | note);
                    log;
                    sample;
                    syslog;
                }
            }
        }
    }
    filter filter-name {
        accounting-profile [ profile-names ];
        interface-specific;
        term term-name {
            filter filter-name;
            from {
                ... statements in Common IP Firewall Match Conditions on page 323 AND
                statements in Common IPv4 Firewall Match Conditions on page 324 ...
            }
            then {
                ... statements in Common Firewall Actions on page 321 AND
                statements in Common IP Firewall Actions on page 322 AND
                statements in Common IPv4 and IPv6 Firewall Actions on page 322 ...
            }
        }
    }
    prefix-action name {
        count;
        destination-prefix-length prefix-length;
    }
}

```

```
filter-specific;
policer policer-name;
source-prefix-length prefix-length;
subnet-prefix-length prefix-length;
}
service-filter filter-name {
  term term-name {
    from {
      address {
        ip-prefix</prefix-length>;
      }
      (ah-spi [ values ] | ah-spi-except [ values ]);
      destination-address {
        ip-prefix</prefix-length>;
      }
      (destination-port [ port-names ] | destination-port-except [ port-names ]);
      destination-prefix-list {
        list-name;
      }
      (esp-spi [ values ] | esp-spi-except [ values ]);
      first-fragment;
      fragment-flags flag;
      (fragment-offset [ offsets ] | fragment-offset-except [ offsets ]);
      (interface-group [ group-names ] | interface-group-except [ group-names ]);
      (ip-options [ option-names ] | ip-options-except [ option-names ]);
      is-fragment;
      (loss-priority [ priorities ] | loss-priority-except [ priorities ]);
      (port [ port-names ] | port-except [ port-names ]);
      prefix-list {
        list-name;
      }
      (protocol [ protocol-names ] | protocol-except [ protocol-names ]);
      source-address {
        ip-prefix</prefix-length>;
      }
      (source-port [ port-names ] | source-port-except [ port-names ]);
      source-prefix-list {
        list-name;
      }
      tcp-flags flag-name;
    }
    then {
      count counter-name;
      log;
      port-mirror;
      sample;
      (service | skip);
    }
  }
}
simple-filter filter-name {
  term term-name {
    from {
      destination-address ip-prefix</prefix-length>;
      destination-port port-name;
      forwarding-class [ class-names ];
    }
  }
}
```

```

        protocol protocol-name;
        source-address ip-prefix</prefix-length>;
        source-port port-name;
    }
    then {
        forwarding-class class-name;
        loss-priority (high | low | medium-high | medium-low);
        policer policer-name;
    }
}
}
}
}

firewall {
    family inet6 {
        dialer-filter filter-name {
            accounting-profile [ profile-names ];
            term term-name {
                from {
                    ... statements in Common IP Firewall Match Conditions on page 323 PLUS ...
                    (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
                    ... BUT NOT ...
                    (destination-class [ class-names ] |
                     destination-class-except [ class-names ]); # NOT valid at this level
                    (forwarding-class [ class-names ] |
                     forwarding-class-except [ class-names ]); # NOT valid at this level
                    interface interface-name; # NOT valid at this level
                    (interface-group [ group-names ] | interface-group-except [ group-names ]); #
                     NOT valid at this level
                    (loss-priority [ priorities ] | loss-priority-except [ priorities ]); # NOT valid at
                     this level
                    service-filter-hit; # NOT valid at this level
                    (source-class [ class-names ] | source-class-except [ class-names ]); # NOT
                     valid at this level
                    tcp-established; # NOT valid at this level
                    tcp-flags flag; # NOT valid at this level
                    tcp-initial; # NOT valid at this level
                }
                then {
                    (ignore | note);
                    log;
                    sample;
                    syslog;
                }
            }
        }
    }
    filter filter-name {
        accounting-profile [ profile-names ];
        interface-specific;
        term term-name {
            filter filter-name;
            from {
                ... statements in Common IP Firewall Match Conditions on page 323 PLUS ...
                (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
                (traffic-class [ code-point-values ] | traffic-class-except [ code-point-values ]);
            }
        }
    }
}

```

```

    }
    then {
        ... statements in Common Firewall Actions on page 321 AND
        statements in Common IP Firewall Actions on page 322 PLUS ...
        (accept | discard | reject <address-unreachable | administratively-prohibited |
        beyond-scope | fragmentation-needed | no-route | port-unreachable |
        tcp-reset>);
    }
}
}
service-filter filter-name {
    term term-name {
        from {
            address {
                ip-prefix </prefix-length>;
            }
            (ah-spi [ values ] | ah-spi-except [ values ]);
            destination-address {
                ip-prefix </prefix-length>;
            }
            (destination-port [ port-names ] | destination-port-except [ port-names ]);
            destination-prefix-list {
                list-name;
            }
            (esp-spi [ values ] | esp-spi-except [ values ]);
            (interface-group [ group-names ] | interface-group-except [ group-names ]);
            (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
            (port [ port-names ] | port-except [ port-names ]);
            prefix-list {
                list-name;
            }
            source-address {
                ip-prefix </prefix-length>;
            }
            (source-port [ port-names ] | source-port-except [ port-names ]);
            source-prefix-list {
                list-name;
            }
            tcp-flags flag-name;
        }
        then {
            count counter-name;
            log;
            port-mirror;
            sample;
            (service | skip);
        }
    }
}
}
}

```

**Related Documentation** • [Notational Conventions Used in Junos OS Configuration Hierarchies](#)

## dscp (Multifield Classifier)

<b>Syntax</b>	<code>dscp [0   <i>value</i>];</code>
<b>Hierarchy Level</b>	[edit firewall family <i>family-name</i> filter <i>filter-name</i> term <i>term-name</i> <b>then</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 7.4.
<b>Description</b>	<p>For M320 and T Series routers, set the DSCP field of incoming or outgoing packets to <b>000000</b>. On the same packets, you can use a behavior aggregate (BA) classifier and a rewrite rule to rewrite the MPLS EXP field.</p> <p>For MX Series routers with MPCs and EX Series switches, the DSCP field can be set from a numeric range.</p> <p>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.</p>
<b>Options</b>	<b>value</b> —For MX Series routers with MPCs, specify the field of incoming or outgoing packets in the range from <b>0</b> through <b>63</b> .
<b>Required Privilege Level</b>	firewall—To view this statement in the configuration. firewall-control—To add this statement to the configuration.
<b>Related Documentation</b>	• <a href="#">Applying Tricolor Marking Policers to Firewall Filters on page 107</a>

## family (Multifield Classifier)

---

<b>Syntax</b>	<pre>family <i>family-name</i> {     filter <i>filter-name</i> {         term <i>term-name</i> {             ... <i>term_configuration</i> ...         }     } }</pre>
<b>Hierarchy Level</b>	[edit firewall]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Configure a firewall filter for IP version 4 (IPv4) or IP version 6 (IPv6) traffic.
<b>Options</b>	<p><i>family-name</i>—Protocol family:</p> <ul style="list-style-type: none"><li>• <b>ccc</b>—Circuit cross-connect parameters</li><li>• <b>inet</b>—IPv4 parameters</li><li>• <b>inet6</b>—IPv6 protocol parameters</li><li>• <b>iso</b>—OSI ISO protocol parameters</li><li>• <b>mlppp</b>—Multilink PPP protocol parameters</li><li>• <b>mpls</b>—MPLS protocol parameters</li><li>• <b>tcc</b>—Translational cross-connect parameters</li><li>• <b>vpls</b>—Virtual private LAN service parameters.</li></ul> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li></ul>

## filter (Configuring)

<b>Syntax</b>	<pre>filter <i>filter-name</i> {     accounting-profile <i>name</i>;     enhanced-mode;     fast-lookup-filter;     filter-list-template;     interface-shared;     interface-specific;     physical-interface-filter;     promote gre-key;     term <i>term-name</i> {         ... term configuration ...     } }</pre>
<b>Hierarchy Level</b>	<p>[edit firewall family <i>family-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> firewall family <i>family-name</i>]</p>
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Logical systems support introduced in Junos OS Release 9.3.</p> <p><b>physical-interface-filter</b> statement introduced in Junos OS Release 9.6.</p> <p>Support for the <b>interface-shared</b> statement introduced in Junos OS Release 12.2.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p>
<b>Description</b>	Configure firewall filters.
<b>Options</b>	<p><b><i>filter-name</i></b>—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 <b>__.*__</b> (beginning and ending with underscores) or <b>__.*</b> (beginning with an underscore).</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>firewall—To view this statement in the configuration.</p> <p>firewall-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Guidelines for Configuring Firewall Filters</i></li> <li>• <i>Guidelines for Applying Firewall Filters</i></li> <li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li> <li>• <i>Using Multifield Classifiers to Set Packet Loss Priority</i></li> <li>• <i>simple-filter (Configuring)</i></li> </ul>

## forwarding-class (Multifield Classifiers)

---

<b>Syntax</b>	<code>forwarding-class <i>class-name</i>;</code>
<b>Hierarchy Level</b>	<code>[edit firewall family <i>family-name</i> filter <i>filter-name</i> term <i>term-name</i> <b>then</b>]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Set the forwarding class of incoming packets.
<b>Options</b>	<i>class-name</i> —Name of the forwarding class.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li></ul>

## from

---

<b>Syntax</b>	<pre>from {   applications [ <i>application-name</i> ];   application-sets [ <i>set-name</i> ];   destination-address <i>address</i>;   source-address <i>address</i>; }</pre>
<b>Hierarchy Level</b>	<code>[edit services cos rule <i>rule-name</i> term <i>term-name</i>]</code>
<b>Release Information</b>	Statement introduced in Junos OS Release 8.1.
<b>Description</b>	Specify input conditions for a CoS term.
<b>Options</b>	The remaining statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring CoS Rule Sets</a></li></ul>

## loss-priority (Firewall Filter)

---

<b>Syntax</b>	loss-priority (high   low);
<b>Hierarchy Level</b>	[edit firewall family <i>family-name</i> filter <i>filter-name</i> term <i>term-name</i> <b>then</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Set the loss priority of incoming packets.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li></ul>

## loss-priority (Simple Firewall Filter)

---

<b>Syntax</b>	loss-priority (high   low   medium);
<b>Hierarchy Level</b>	[edit firewall family <i>family-name</i> simple-filter <i>filter-name</i> term <i>term-name</i> <b>then</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 7.6.
<b>Description</b>	Set the loss priority of incoming packets.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li></ul>

## term (Simple Filter)

---

Syntax	<pre>term <i>term-name</i> {     from {         <i>match-conditions</i>;     }     then {         forwarding-class <i>class-name</i>;         loss-priority (high   low   medium);     } }</pre>
Hierarchy Level	[edit firewall family inet simple-filter <i>filter-name</i> ]
Release Information	Statement introduced in Junos OS Release 7.6.
Description	Define a simple filter term.
Options	<p><b>from</b>—Match packet fields to values. If the <b>from</b> option is not included, all packets are considered to match and the actions and action modifiers in the <b>then</b> statement are taken.</p> <p><b>match-conditions</b>—One or more conditions to use to make a match. The conditions are described in the <i>Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices</i>.</p> <p><b>term-name</b>—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 (" ").</p> <p><b>then</b>—Actions to take on matching packets. If the <b>then</b> option is not included and a packet matches all the conditions in the <b>from</b> statement, the packet is accepted. For CoS, only the actions listed are allowed. These statements are explained separately.</p>
Usage Guidelines	See <a href="#">“Configuring Multifield Classifiers” on page 70</a> ; for a general discussion of this statement, see the <i>Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices</i> .
Required Privilege Level	firewall—To view this statement in the configuration. firewall-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"><li>• <i>Multifield Classification</i></li><li>• <i>Simple Filter Overview</i></li><li>• <i>Firewall Filter Match Conditions for IPv4 Traffic</i></li><li>• <i>Firewall Filter Match Conditions for IPv6 Traffic</i></li></ul>

## then (Services CoS)

<b>Syntax</b>	<pre> then {   application-profile <i>profile-name</i>;   dscp (<i>alias</i>   <i>bits</i>);   forwarding-class <i>class-name</i>;   syslog;   (reflexive   reverse) {     application-profile <i>profile-name</i>;     dscp (<i>alias</i>   <i>bits</i>);     forwarding-class <i>class-name</i>;     syslog;   } } </pre>
<b>Hierarchy Level</b>	[edit services cos rule <i>rule-name</i> term <i>term-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.1.
<b>Description</b>	<p>Define the CoS term actions.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>Configuring Actions in a CoS Rule</li> <li>Configuring Actions in CoS Rules</li> </ul>

## [edit interfaces] Hierarchy Level

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

```

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

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

```

```
no-gratuitous-arp-request;

traceoptions {
    flag flag;
}
(traps | no-traps);
unit logical-unit-number {
    accounting-profile name;
    bandwidth rate;
    description text;
    disable;
    encapsulation type;
    family inet {
        accounting {
            destination-class-usage;
            source-class-usage {
                input;
                output;
            }
        }
    }
    address ipv4-address {
        arp ip-address (mac | multicast-mac) mac-address <publish>;
        broadcast address;
        preferred;
        primary;
        vrrp-group group-id {
            (accept-data | no-accept-data);
            advertise-interval seconds;
            advertisements-threshold number;
            authentication-key key;
            authentication-type authentication;
            fast-interval milliseconds;
            (preempt | no-preempt) {
                hold-time seconds;
            }
            priority number;
            track {
                interface interface-name {
                    bandwidth-threshold bits-per-second priority-cost priority;
                    priority-cost priority;
                }
                priority-hold-time seconds;
                route prefix/prefix-length routing-instance instance-name priority-cost priority;
            }
            virtual-address [ addresses ];
            vrrp-inherit-from vrrp-group;
        }
    }
}
filter {
    input filter-name;
    output filter-name;
}
mtu bytes;
no-neighbor-learn;
no-redirects;
```

```

primary;
rpf-check {
    fail-filter filter-name;
    mode {
        loose;
    }
}
targeted-broadcast {
    forward-and-send-to-re;
    forward-only;
}
}
family inet6 {
    accounting {
        destination-class-usage;
        source-class-usage {
            input;
            output;
        }
    }
}
address address {
    eui-64;
    ndp ip-address (mac | multicast-mac) mac-address <publish>;
    preferred;
    primary;
    vrrp-inet6-group group-id {
        accept-data | no-accept-data;
        advertisements-threshold number;
        authentication-key key;
        authentication-type authentication;
        fast-interval milliseconds;
        inet6-advertise-interval milliseconds;
        preempt | no-preempt {
            hold-time seconds;
        }
        priority number;
        track {
            interface interface-name {
                bandwidth-threshold bandwidth priority-cost number;
                priority-cost number;
            }
            priority-hold-time seconds;
            route ip-address/mask routing-instance instance-name priority-cost cost;
        }
        virtual-inet6-address [addresses];
        virtual-link-local-address ipv6-address;
        vrrp-inherit-from {
            active-group group-number;
            active-interface interface-name;
        }
    }
}
}
(dad-disable | no-dad-disable);
filter {
    input filter-name;
    output filter-name;
}

```

```
    }
    mtu bytes;
    nd6-stale-time seconds;
    no-neighbor-learn;
    no-redirects;
    policer {
        input policer-name;
        output policer-name;
    }
    rpf-check {
        fail-filter filter-name;
        mode {
            loose;
        }
    }
}
family iso {
    address interface-address;
    mtu bytes;
}
family mpls {
    filter {
        input filter-name;
        output filter-name;
    }
    mtu bytes;
    policer {
        input policer-name;
        output policer-name;
    }
}
native-inner-vlan-id vlan-id;
proxy-arp (restricted | unrestricted);
(traps | no-traps);
vlan-id-list [vlan-id's];
vlan-id-range [vlan-id-range];
}
}
traceoptions {
    file <filename> <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
    flag flag <disable>;
    no-remote-trace;
}
}

interfaces {
    interface-name {
        disable;
        accounting-profile name;
        aggregated-ether-options {
            ethernet-switch-profile {
                tag-protocol-id [ hexadecimal-identifiers ];
            }
        }
        (flow-control | no-flow-control);
        lacp {
```

```

    (active | passive);
    admin-key key;
    fast-failover;
    link-protection {
        disable;
        (revertive | non-revertive);
    }
    periodic (fast | slow);
    system-id mac-address;
    system-priority priority;
}
(link-protection | no-link-protection);
link-speed (100m | 1g | 8g | 10g | 40g | 50g | 80g | 100g | oc192);
logical-interface-fpc-redundancy;
(loopback | no-loopback);
mc-ae {
    chassis-id chassis-id;
    events {
        iccp-peer-down {
            force-icl-down;
            prefer-status-control-active;
        }
    }
    mc-ae-id mc-ae-id;
    mode (active-active | active-standby);
    redundancy-group group-id;
    status-control (active | standby);
}
minimum-links number;
rebalance-periodic {
    start-time time;
    interval number;
}
source-address-filter {
    mac-address;
}
(source-filtering | no-source-filtering);
}
auto-configure {
    remove-when-no-subscribers;
    stacked-vlan-ranges {
        access-profile profile-name;
        authentication {
            password password-string;
            username-include {
                circuit-type;
                delimiter delimiter-character;
                domain-name domain-name-string;
                interface-name;
                mac-address;
                option-82 ( circuit-id | remote-id);
                radius-realm radius-realm-string;
                user-prefix user-prefix-string;
            }
        }
    }
    dynamic-profile profile-name {

```

```

        accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
        ranges (any | low-tag-high-tag), (any | low-tag-high-tag);
    }
}
vlan-ranges {
    access-profile profile-name;
    authentication {
        password password-string;
        username-include {
            circuit-type;
            delimiter delimiter-character;
            domain-name domain-name-string;
            interface-name;
            mac-address;
            option-82;
            radius-realm radius-realm-string;
            user-prefix user-prefix-string;
        }
    }
    dynamic-profile profile-name {
        accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
        ranges (any | low-tag)–(any | high-tag);
    }
}
override tag vlan-tag dynamic-profile profile name;
}
encapsulation (ethernet-bridge | ethernet-vpls | extended-vlan-bridge |
    extended-vlan-vpls | flexible-ethernet-services | vlan-vpls);
ether-options {
    802.3ad {
        aex;
        (backup | primary);
        lacp {
            force-up;
            port-priority
        }
    }
}
asynchronous-notification;
(auto-negotiation | no-auto-negotiation);
ethernet-switch-profile {
    ethernet-policer-profile {
        input-priority-map {
            ieee802.1p premium [ values ];
        }
        output-priority-map {
            classifier {
                premium {
                    forwarding-class class-name {
                        loss-priority (high | low);
                    }
                }
            }
        }
    }
}
policer cos-policer-name {
    aggregate {
        bandwidth-limit bps;
    }
}

```

```

        burst-size-limit bytes;
    }
    premium {
        bandwidth-limit bps;
        burst-size-limit bytes;
    }
}
tag-protocol-id;
}
(mac-learn-enable | no-mac-learn-enable);
}
(flow-control | no-flow-control);
ignore-l3-incompletes;
link-mode (automatic | full-duplex | half-duplex);
(loopback | no-loopback);
keepalives <interval seconds> <down-count number> <up-count number>;
speed (1g | 10m | 100m | 10m-100m | auto-negotiation);
source-address-filter {
    mac-address;
}
source-filtering | no-source-filtering;
}
flexible-vlan-tagging;
(gratuitous-arp-reply | no-gratuitous-arp-reply);
hold-time (up milliseconds | down milliseconds);
interface-transmit-statistics;
(keepalives <down-count number> <interval seconds> <up-count number> |
no-keepalives);
layer2-policer {
    apply-groups [ group-names ];
    apply-groups-except [ group-names ];
}
link-mode (automatic | full-duplex);
mac mac-address;
mtu bytes;
multi-chassis-protection peer-ip-address {
    interface interface-name;
}
native-vlan-id number;
no-gratuitous-arp-request;
optics-options {
    alarm low-light-alarm {
        (link-down | syslog);
    }
    warning low-light-warning {
        (link-down | syslog);
    }
}
wavelength nm;
}
passive-monitor-mode;
per-unit-scheduler;
speed (10m | 100m | 1g | auto | oc3 | oc12 | oc48);
stacked-vlan-tagging;
traceoptions {
    flag flag;
}

```

```
transmit-bucket {  
    overflow discard;  
    rate percentage;  
    threshold bytes;  
}  
(traps | no-traps);  
unidirectional;  
vlan-tagging;  
}
```

```
interface-name {  
    unit logical-unit-number {  
        disable;  
        accept-source-mac {  
            mac-address mac-address {  
                policer {  
                    input policer-name;  
                    output policer-name;  
                }  
            }  
        }  
    }  
    accounting-profile name;  
    advisory-options {  
        downstream-rate rate;  
        upstream-rate rate;  
    }  
    arp-resp (restricted|unrestricted);  
    bandwidth rate;  
    clear-dont-fragment-bit;  
    copy-tos-to-outer-ip-header;  
    demux-destination family;  
    encapsulation (vlan-bridge | vlan-vpls);  
    epd-threshold cells plp1 cells;  
    filter filter-name;  
    inner-vlan-id-range start start-id end end-id;  
    input-vlan-map {  
        (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);  
        inner-tag-protocol-id tpid;  
        inner-vlan-id number;  
        tag-protocol-id tpid;  
        vlan-id number;  
    }  
    interface-shared-with psd numerical-index;  
    layer2-policer {  
        input-hierarchical-policer policer-name;  
        input-policer policer-name;  
        input-three-color policer-name;  
        output-policer policer-name;  
        output-three-color policer-name;  
    }  
    multi-chassis-protection peer-ip-address {  
        interface interface-name;  
    }  
    native-inner-vlan-id number;  
    output-vlan-map {
```

```

(pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
inner-tag-protocol-id tpid;
inner-vlan-id number;
tag-protocol-id tpid;
vlan-id number;
}
peer-interface interface-name;
peer-unit unit-number;
plp-to-clp;
proxy-arp <restricted | unrestricted>;
rpm {
    (client | server);
    twamp-server;
}
swap-by-poppush;
vlan-id number;
vlan-id-list [ vlan-id vlan-id-vlan-id ];
vlan-id-range number-number;
vlan-tags (inner <tpid.>vlan-id | inner-list [ vlan-id vlan-id-vlan-id ] |
    inner-range <tpid.>vlan-id-vlan-id) outer <tpid.>vlan-id;
}

unit logical-unit-number {
    family ethernet-switching {
        filter {
            group filter-group-number;
            (input filter-name | input-list [ filter-names ]);
            (output filter-name | output-list [ filter-names ]);
            (inner-vlan-id-list [ vlan-ids ] | vlan-id number | vlan-id-list [ number
                number-number ]);
            interface-mode (access | trunk);
            policer {
                input policer-name;
                output policer-name;
            }
            vlan-rewrite {
                translate old-vlan-id new-vlan-id;
            }
            vlan {
                members [ all vlan-identifiers ];
            }
        }
    }
    family inet {
        filter {
            group filter-group-number;
            (input filter-name | input-list [ filter-names ]);
            (output filter-name | output-list [ filter-names ]);
        }
        input-hierarchical-policer policer-name;
        mac-validate (loose | strict);
        mtu bytes;
        no-neighbor-learn;
        no-redirects;
        policer {
            arp policer-template-name;
            input policer-name;

```

```

        output policer-name;
    }
    primary;
    receive-options-packets;
    receive-ttl-exceeded;
    rpf-check {
        fail-filter filter-name;
        mode loose;
    }
    sampling {
        (input | output | input output);
    }
    simple-filter {
        input filter-name;
    }
    targeted-broadcast {
        forward-and-send-to-re;
        forward-only;
    }
    unnumbered-address interface-name <destination address>
        <destination-profile profile-name> <preferred-source-address address>;
}

family inet6 {
    address ipv6-address {
        destination destination-address;
        eui-64;
        ndp ipv6-address <l2-interface interface-name> <(mac mac-address |
            multicast-mac multicast-mac-address) <publish>>;
        preferred;
        primary;
        vrrp-inet6-group group-number {
            (accept-data | no-accept-data);
            fast-interval milliseconds;
            inet6-advertise-interval seconds;
            (no-preempt; | ... the following preempt statement ...)
            preempt {
                hold-time seconds;
            }
            priority number;
            track {
                interface interface-name {
                    bandwidth-threshold bits-per-second priority-cost priority;
                    priority-cost priority;
                }
                priority-hold-time seconds;
                route ip-address-prefix/prefix-length routing-instance instance-name
                    priority-cost priority;
            }
        }
        virtual-inet6-address [ addresses ];
        virtual-link-local-address ipv6-address;
        vrrp-inherit-from {
            active-group group-number;
            active-interface interface-name;
        }
    }
}

```

```

    }
  }
  (dad-disable | no-dad-disable);
  filter {
    group filter-group-number;
    (input filter-name | input-list [ filter-names ]);
    (output filter-name | output-list [ filter-names ]);
  }
  input-hierarchical-policer policer-name;
  mtu bytes;
  nd6-stale-time seconds;
  no-neighbor-learn;
  policer {
    input policer-name;
    output policer-name;
  }
  rpf-check {
    fail-filter filter-name;
    mode loose;
  }
  sampling {
    (input | output | input output);
  }
  unnumbered-address interface-name preferred-source-address address;
}

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

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

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

family vpls {

```

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

**Related  
Documentation**

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## filter (Applying to an Interface)

<b>Syntax</b>	<pre>filter {     input <i>filter-name</i>;     output <i>filter-name</i>; }</pre>
<b>Hierarchy Level</b>	[edit interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i> <b>family</b> <i>family</i> ], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i> <b>family</b> <i>family</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Apply a filter to an interface. You can also use filters for encrypted traffic. When you configure filters, you can configure the family <b>inet</b> , <b>inet6</b> , <b>mpls</b> , or <b>vppls</b> only.
<b>Options</b>	<p><b>input <i>filter-name</i></b>—Name of one filter to evaluate when packets are received on the interface.</p> <p><b>output <i>filter-name</i></b>—Name of one filter to evaluate when packets are transmitted on the interface.</p>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">simple-filter on page 468</a></li> <li>• <i>Applying Firewall Filter Tricolor Marking Policers to Interfaces</i></li> <li>• <i>Example: Classifying Packets Based on Their Destination Address</i></li> <li>• <a href="#">Example: Configuring and Verifying a Complex Multifield Filter on page 74</a></li> <li>• <i>Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets</i></li> <li>• <a href="#">Example: Configuring a Simple Filter on page 254</a></li> <li>• <a href="#">Example: Configuring a Logical Bandwidth Policer on page 89</a></li> <li>• <a href="#">Configuring Two-Color Policers and Shaping Rate Changes on page 86</a></li> </ul>

## simple-filter (Applying to an Interface)

---

<b>Syntax</b>	<code>simple-filter {     input <i>filter-name</i>; }</code>
<b>Hierarchy Level</b>	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>family</b> inet], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>family</b> inet]
<b>Release Information</b>	Statement introduced in Junos OS Release 7.6.
<b>Description</b>	Apply a simple filter to an interface. You can apply simple filters to the family <b>inet</b> only, and only in the input direction.
<b>Options</b>	<b>input <i>filter-name</i></b> —Name of one filter to evaluate when packets are received on the interface.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li><li>• <a href="#">filter (Applying to an Interface) on page 467</a></li></ul>

# Configuration Statements: RED Drop Profiles

- [\[edit class-of-service\] Hierarchy Level on page 469](#)
- [drop-probability \(Interpolated Value\) on page 473](#)
- [drop-profiles on page 474](#)
- [fill-level \(Interpolated Value\) on page 475](#)
- [fill-level \(Drop Profiles\) on page 476](#)
- [interpolate on page 476](#)

## [\[edit class-of-service\] Hierarchy Level](#)

---

```

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

```

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

```

rewrite-rules {
  (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
  rewrite-rule {
    forwarding-class class-name {
      loss-priority level code-point (alias | bits);
    }
    import (rewrite-rule | default);
  }
}
routing-instances routing-instance-name {
  classifiers {
    dscp (classifier-name | default);
    dscp-ipv6 (classifier-name | default);
    exp (classifier-name | default);
    ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
  }
}
scheduler-maps {
  map-name {
    forwarding-class class-name scheduler scheduler-name;
  }
}
schedulers {
  scheduler-name {
    adjust-minimum value;
    adjust-percent value;
    buffer-size (exact | percent percentage | remainder);
    drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
      protocol any;
    excess-priority (high | low | medium-high | medium-low);
    excess-rate (percent percentage | proportion proportion);
    priority (high | low | medium-high | medium-low | strict-high);
    shaping-rate (bps | percent percentage | burst-size size);
    transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
  }
}
traceoptions {
  file <files number> <match regular-expression> <size maximum-file-size>
    <world-readable | no-world-readable>;
  flag flag;
  no-remote-trace;
}
traffic-control-profiles {
  profile-name {
    adjust-minimum rate;
    delay-buffer-rate (bps | cps cps | percent percentage);
    excess-rate (percent percentage | proportion value);
    guaranteed-rate (bps | percent percentage) <burst-size bytes>;
    overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
    scheduler-map map-name;
    shaping-rate (bps | percent percentage) <burst-size bytes>;
  }
}
tri-color;
}

```

```

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

```

```

        excess-bandwidth-share (equal | proportional value);
        input-excess-bandwidth-share (equal | proportional value);
        input-traffic-control-profile profile-name;
        input-traffic-control-profile-remaining profile-name;
        internal-node;
        output-traffic-control-profile profile-name;
        output-traffic-control-profile-remaining profile-name;
    }
}

```

**Related Documentation** • *Notational Conventions Used in Junos OS Configuration Hierarchies*

## drop-probability (Interpolated Value)

<b>Syntax</b>	drop-probability [ <i>values</i> ];
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">drop-profiles</a> <i>profile-name</i> <a href="#">interpolate</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS 11.4 for EX Series switches.
<b>Description</b>	Define up to 64 values for interpolating drop probabilities on Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers. On EX Series switches, this statement is supported only on the EX9200 switch, EX8200 standalone switches, and EX8200 Virtual Chassis.
<b>Options</b>	<b><i>percentage</i></b> —The probability (expressed in percentage) for a packet to be dropped from the queue. <b>Range:</b> 0 through 100
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	• <i>Default Drop Profile</i>

## drop-profiles

---

<b>Syntax</b>	<pre>drop-profiles {   profile-name {     fill-level percentage drop-probability percentage;     interpolate {       drop-probability [values];       fill-level [values]     }   } }</pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS 11.4 for EX Series switches.
<b>Description</b>	<p>Define drop profiles for RED.</p> <p>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.</p>
<b>Options</b>	<p><i>profile-name</i>—Name of the drop profile.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 205</a></li></ul>

## fill-level (Interpolated Value)

---

<b>Syntax</b>	fill-level [ <i>values</i> ];
<b>Hierarchy Level</b>	[edit class-of-service <b>drop-profiles</b> <i>profile-name</i> interpolate]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS 11.4 for EX Series switches.
<b>Description</b>	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.
<b>Options</b>	<b>values</b> —Data points for mapping queue fill percentage. <b>Range:</b> 0 through 100 <b>Default:</b> 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.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 200</a></li> <li>• <a href="#">Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 205.</a></li> </ul>

## fill-level (Drop Profiles)

---

<b>Syntax</b>	<code>fill-level <i>percentage</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">drop-profiles</a> <i>profile-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS 11.4 for EX Series switches.
<b>Description</b>	When configuring RED, map the fullness of a queue to a drop probability.
<b>Options</b>	<b><i>percentage</i></b> —How full the queue is, expressed as a percentage. You configure the <b>fill-level</b> and <b>drop-probability</b> statements in pairs. To specify multiple fill levels, include multiple <b>fill-level</b> and <b>drop-probability</b> statements. The values you assign to each statement pair must increase relative to the previous pair's values. This is shown in the segmented graph in " <a href="#">Managing Congestion Using RED Drop Profiles and Packet Loss Priorities</a> " on page 200. <b>Range:</b> 0 through 100 percent
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 200</a></li><li>• <a href="#">Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 205</a></li></ul>

## interpolate

---

<b>Syntax</b>	<pre>interpolate {     <a href="#">drop-probability</a> [<i>values</i>];     <a href="#">fill-level</a> [<i>values</i>]; }</pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">drop-profiles</a> <i>profile-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS 11.4 for EX Series switches.
<b>Description</b>	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.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• See <a href="#">Defining Packet Drop Behavior by Configuring RED Drop Profiles on page 205</a>.</li></ul>

## CHAPTER 27

# Configuration Statements: Rewriting Packet Header Information

- [\[edit class-of-service\] Hierarchy Level](#) on page 478
- [code-point](#) on page 482
- [default \(CoS Host Outbound Traffic\)](#) on page 482
- [dscp \(Rewrite Rules\)](#) on page 483
- [dscp \(Rewrite Rules on Physical Interface\)](#) on page 484
- [dscp-ipv6 \(CoS Rewrite Rules\)](#) on page 485
- [exp](#) on page 486
- [exp-push-push-push](#) on page 487
- [exp-swap-push-push](#) on page 488
- [forwarding-class \(BA Classifiers\)](#) on page 489
- [frame-relay-de \(Defining Loss Priority Maps\)](#) on page 490
- [host-outbound-traffic \(Class-of-Service\)](#) on page 491
- [ieee-802.1 \(Rewrite Rules on Logical Interface\)](#) on page 492
- [ieee-802.1 \(Host Outbound Traffic\)](#) on page 493
- [ieee-802.1 \(Rewrite Rules on Physical Interface\)](#) on page 493
- [ieee-802.1ad](#) on page 494
- [import \(Rewrite Rules\)](#) on page 495
- [inet-precedence \(CoS Rewrite Rules\)](#) on page 495
- [inet-precedence \(Rewrite Rules on Physical Interface\)](#) on page 496
- [interfaces](#) on page 497
- [loss-priority \(BA Classifiers\)](#) on page 499
- [loss-priority-maps](#) on page 500
- [loss-priority-maps \(Assigning to an Interface\)](#) on page 501
- [protocol \(Rewrite Rules\)](#) on page 502
- [rewrite-rules \(CoS Host Outbound Traffic\)](#) on page 503
- [rewrite-rules \(Definition\)](#) on page 504

- [rewrite-rules \(Interfaces\)](#) on page 505
- [rewrite-rules \(Physical Interfaces\)](#) on page 506
- [unit](#) on page 507
- [vlan-tag](#) on page 508

---

## [edit class-of-service] Hierarchy Level

---

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

```

        discard;
        lsp-next-hop [ lsp-regular-expressions ];
        next-hop [ next-hop-names ];
        non-lsp-next-hop;
    }
}
}
fragmentation-maps {
    map-name {
        forwarding-class class-name {
            drop-timeout milliseconds;
            fragment-threshold bytes;
            multilink-class number;
            no-fragmentation;
        }
    }
}
host-outbound-traffic {
    dscp-code-point value;
    forwarding-class class-name;
    ieee-802.1 {
        default value;
        rewrite-rules;
    }
    tcp {
        raise-internet-control-priority;
    }
}
interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
}
}
restricted-queues {
    forwarding-class class-name queue-number;
}
rewrite-rules {
    (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
    rewrite-rule {
        forwarding-class class-name {
            loss-priority level code-point (alias | bits);
        }
        import (rewrite-rule | default);
    }
}
routing-instances routing-instance-name {
    classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
    }
}
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}

```

```

    }
  }
  schedulers {
    scheduler-name {
      adjust-minimum value;
      adjust-percent value;
      buffer-size (exact | percent percentage | remainder);
      drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
        protocol any;
      excess-priority (high | low | medium-high | medium-low);
      excess-rate (percent percentage | proportion proportion);
      priority (high | low | medium-high | medium-low | strict-high);
      shaping-rate (bps | percent percentage | burst-size size);
      transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
    }
  }
  traceoptions {
    file <files number> <match regular-expression> <size maximum-file-size>
      <world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
  }
  traffic-control-profiles {
    profile-name {
      adjust-minimum rate;
      delay-buffer-rate (bps | cps cps | percent percentage);
      excess-rate (percent percentage | proportion value);
      guaranteed-rate (bps | percent percentage) <burst-size bytes>;
      overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
      scheduler-map map-name;
      shaping-rate (bps | percent percentage) <burst-size bytes>;
    }
  }
  tri-color;
}

class-of-service {
  interfaces {
    interface-name {
      excess-bandwidth-share (equal | proportional value);
      input-excess-bandwidth-share (equal | proportional value);
      input-scheduler-map map-name;
      input-shaping-rate bps;
      input-traffic-control-profile profile-name;
      output-forwarding-class-map map-name;
      output-traffic-control-profile profile-name;
      scheduler-map map-name;
      scheduler-map-chassis (map-name | derived);
      shaping-rate bps;
      unit (logical-unit-number | *) {
        classifiers {
          dscp (classifier-name | default) {
            family [ inet mpls ];
          }
          dscp-ipv6 (classifier-name | default) {
            family [ inet mpls ];
          }
        }
      }
    }
  }
}

```

```

    }
    exp (classifier-name | default);
    ieee-208.1 (classifier-name | default) <vlan-tag (inner | outer)>;
    ieee-208.1ad (classifier-name | default);
    inet-precedence (classifier-name | default);
  }
  forwarding-class class-name;
  input-scheduler-map map-name;
  input-shaping-rate bps;
  input-traffic-control-profile profile-name shared-instance instance-name;
  loss-priority-maps {
    (map-name | default);
  }
  loss-priority-rewrites {
    (map-name | default);
  }
  output-forwarding-class-map map-name;
  output-traffic-control-profile profile-name shared-instance instance-name;
  rewrite-rules {
    dscp (rule-name | default) <protocol mpls>;
    dscp-ipv6 (rule-name | default);
    exp (rule-name | default) <protocol [ mpls-any | mpls-inet-both |
      mpls-inet-both-non-vpn ]>;
    exp-push-push-push default;
    exp-swap-push-push default;
    ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
    ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
    inet-precedence (rewrite-name | default) <protocol mpls>;
  }
  scheduler-map map-name;
  shaping-rate bps;
  translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
    to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
}
}
interface-set interface-set-name {
  excess-bandwidth-share (equal | proportional value);
  input-excess-bandwidth-share (equal | proportional value);
  input-traffic-control-profile profile-name;
  input-traffic-control-profile-remaining profile-name;
  internal-node;
  output-traffic-control-profile profile-name;
  output-traffic-control-profile-remaining profile-name;
}
}
}

```

#### Related Documentation

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## code-point

---

<b>Syntax</b>	<code>code-point [ <i>aliases</i> ] [ <i>bit-patterns</i> ];</code>
<b>Hierarchy Level</b>	[edit class-of-service rewrite-rules <i>type</i> <i>rewrite-name</i> forwarding-class <i>class-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Specify one or more code-point aliases or bit sets for association with a forwarding class.
<b>Options</b>	<i>aliases</i> —Name of each alias.  <i>bit-patterns</i> —Value of the code-point bits, in decimal form.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li></ul>

## default (CoS Host Outbound Traffic)

---

<b>Syntax</b>	<code>default <i>value</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service host-outbound-traffic <a href="#">ieee-802.1p</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.3.
<b>Description</b>	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.
<b>Options</b>	<i>value</i> —Three-bit binary number. <b>Range:</b> 000 through 111
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic on page 232</a></li><li>• <a href="#">Rewriting Packet Headers to Ensure Forwarding Behavior on page 212</a></li></ul>

## dscp (Rewrite Rules)

<b>Syntax</b>	<code>dscp (rewrite-name   default) protocol mpls;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>For IPv4 traffic, apply a Differentiated Services (DiffServ) code point (DSCP) rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"> <li>On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li> <li>On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li> </ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules dscp</b>] hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> <li><a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li><a href="#">protocol (Rewrite Rules) on page 502</a></li> <li><a href="#">Rewriting MPLS and IPv4 Packet Headers</a></li> <li><a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## dscp (Rewrite Rules on Physical Interface)

---

<b>Syntax</b>	dscp ( <i>rewrite-name</i>   default);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> <a href="#">rewrite-rules</a>
<b>Release Information</b>	Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access routers.
<b>Description</b>	Associate a rewrite-rules configuration or default mapping with a specific interface.
<b>Options</b>	<b><i>rewrite-name</i></b> —Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules</b> ] hierarchy level.  <b>default</b> —The default mapping.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.

## dscp-ipv6 (CoS Rewrite Rules)

<b>Syntax</b>	<code>dscp-ipv6 (rewrite-name   &lt;default&gt;) protocol mpls;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <a href="#">rewrite-rules</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Support for <b>protocol mpls</b> option introduced in Junos OS Release 10.4R2.
<b>Description</b>	<p>For IPv6 traffic, apply a DSCP rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp-ipv6</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"> <li>On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li> <li>On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li> </ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules dscp-ipv6</b>] hierarchy level.</p> <p><b>default</b>—Default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> <li><a href="#">protocol on page 502</a></li> <li><a href="#">Setting IPv6 DSCP and MPLS EXP Values Independently on page 217</a></li> <li><a href="#">Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 220</a></li> <li><a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li><a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## exp

<b>Syntax</b>	<code>exp (rewrite-name   default) protocol protocol-types;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS Release 12.2. for ACX series
<b>Description</b>	Apply an MPLS experimental (EXP) rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service rewrite-rules exp] hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p>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 <b>mpls-inet-both</b> or <b>mpls-inet-both-non-vpn</b> option at the [edit class-of-service interfaces <i>interface</i> <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> exp <i>rewrite-rule-name</i> protocol] hierarchy level. The IP precedence rewrite rules explanation does not apply to ACX Series Universal Access routers.</p> <p>On interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series 3D Universal Edge Routers and EX Series switches, we highly recommend that you configure the <b>default</b> option when you configure a behavior aggregate (BA) classifier that does not include a specific rewrite rule for MPLS packets. Doing so ensures that MPLS exp value is rewritten according to the BA classifier rules configured for forwarding or packet loss priority. This does not apply to ACX Series Universal Access routers.</p> <p><b>protocol-types</b>—Specify one or more protocol matching criteria:</p> <ul style="list-style-type: none"> <li>• <b>mpls-any</b>—Apply to MPLS packets, write MPLS header only.</li> <li>• <b>mpls-inet-both</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header.</li> <li>• <b>mpls-inet-both-non-vpn</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header for only non VPN traffic.</li> </ul>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</a></li> <li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li> </ul>

- [rewrite-rules \(Definition\) on page 504](#)

## exp-push-push-push

<b>Syntax</b>	exp-push-push-push default;
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	For M Series routers, rewrite the EXP bits of all three labels of an outgoing packet, thereby maintaining CoS of an incoming non-MPLS packet.
<b>Options</b>	<b>default</b> —Apply the default MPLS EXP rewrite table.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</i></li> <li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li> <li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li> <li>• <a href="#">exp on page 377</a></li> <li>• <a href="#">exp-swap-push-push on page 488</a></li> <li>• <a href="#">ieee-802.1 (Rewrite Rules on Logical Interface) on page 379</a></li> <li>• <a href="#">ieee-802.1ad on page 381</a></li> <li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## exp-swap-push-push

---

<b>Syntax</b>	exp-swap-push-push default;
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	For M Series routers, rewrite the EXP bits of all three labels of an outgoing packet, thereby maintaining CoS of an incoming MPLS packet.
<b>Options</b>	<b>default</b> —Apply the default MPLS EXP rewrite table.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</i></li><li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li><li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li><li>• <a href="#">exp on page 377</a></li><li>• <a href="#">exp-push-push-push on page 487</a></li><li>• <a href="#">ieee-802.1 (Rewrite Rules on Logical Interface) on page 379</a></li><li>• <a href="#">ieee-802.1ad on page 381</a></li><li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

---

## forwarding-class (BA Classifiers)

---

<b>Syntax</b>	<code>forwarding-class <i>class-name</i> {     <i>loss-priority level code-points</i> [ <i>aliases</i> ] [ <i>bit-patterns</i> ]; }</code>
<b>Hierarchy Level</b>	[edit class-of-service classifiers <i>type classifier-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Define forwarding class name and option values.
<b>Options</b>	<i>class-name</i> —Name of the forwarding class.  The remaining statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Behavior Aggregate Classifiers on page 36</a></li></ul>

## frame-relay-de (Defining Loss Priority Maps)

---

<b>Syntax</b>	<pre>frame-relay-de <i>name</i> {     loss-priority <i>level</i> <b>code-points</b> [ <i>alias</i>   <i>bits</i> ]; }</pre>
<b>Hierarchy Level</b>	[edit class-of-service loss-priority-maps]
<b>Release Information</b>	Statement introduced in Junos OS Release 11.4.
<b>Description</b>	Define a Frame Relay discard eligibility (DE) bit loss priority map.
<b>Options</b>	<p><b><i>name</i></b>—Name of the loss priority map.</p> <p><b>loss-priority <i>level</i></b>—Level of the loss priority to be applied based on the specified CoS values. The loss priority level can be one of the following:</p> <ul style="list-style-type: none"><li>• <b>high</b>—Packet has high loss priority.</li><li>• <b>low</b>—Packet has low loss priority.</li><li>• <b>medium-high</b>—Packet has medium-high loss priority.</li><li>• <b>medium-low</b>—Packet has medium-low loss priority.</li></ul> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Defining a Custom Frame Relay Loss Priority Map</i></li></ul>

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

<b>Syntax</b>	<pre> host-outbound-traffic {     forwarding-class <i>class-name</i>;     dscp-code-point <i>value</i>;     ieee-802.1 {         default <i>value</i>;         rewrite-rules;     } } </pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced before Junos OS Release 11.4 for EX Series switches.</p> <p>Support for <b>ieee-802.1</b> statement introduced in Junos OS Release 12.3.</p> <p>Support for distributed protocol handler traffic introduced in Junos OS Release 13.2.</p>
<b>Description</b>	Classify and mark host outbound traffic. This statement does not affect transit traffic or incoming traffic.
<b>Default</b>	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.
<b>Options</b>	The remaining statements are explained separately.
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Default Queue Assignments for Routing Engine Sourced Traffic</a></li> <li>• <a href="#">Default DSCP and DSCP IPv6 Classifiers on page 42</a></li> <li>• <a href="#">Changing the Default Queuing and Marking of Host Outbound Traffic on page 137.</a></li> <li>• <a href="#">Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic on page 232</a></li> <li>• <a href="#">Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 232</a></li> </ul>

## ieee-802.1 (Rewrite Rules on Logical Interface)

---

<b>Syntax</b>	ieee-802.1 ( <i>rewrite-name</i>   default) <b>vlan-tag</b> (outer   outer-and-inner);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. <b>vlan-tag</b> statement introduced in Junos OS Release 8.1.
<b>Description</b>	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.
<b>Options</b>	<b>rewrite-name</b> —Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules</b> <b>ieee-802.1</b> ] hierarchy level.  <b>default</b> —The default mapping.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li><li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li><li>• <a href="#">exp on page 377</a></li><li>• <a href="#">exp-push-push-push on page 487</a></li><li>• <a href="#">exp-swap-push-push on page 488</a></li><li>• <a href="#">ieee-802.1ad on page 381</a></li><li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## ieee-802.1 (Host Outbound Traffic)

<b>Syntax</b>	<pre>ieee-802.1 {     default <i>value</i>;     rewrite-rules; }</pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">host-outbound-traffic</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.3.
<b>Description</b>	<p>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.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic on page 232</a></li> <li>• <a href="#">Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 232</a></li> <li>• <a href="#">Rewriting Packet Headers to Ensure Forwarding Behavior on page 212</a></li> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## ieee-802.1 (Rewrite Rules on Physical Interface)

<b>Syntax</b>	<pre>ieee-802.1 (<i>rewrite-name</i>   default) ;</pre>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> ] <a href="#">rewrite-rules</a>
<b>Release Information</b>	Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access routers.
<b>Description</b>	Apply an IEEE-802.1 rewrite rule.
<b>Options</b>	<p><b><i>rewrite-name</i></b>—Name of a <a href="#">rewrite-rules</a> mapping configured at the [edit class-of-service <a href="#">rewrite-rules ieee-802.1</a>] hierarchy level.</p> <p><b>default</b>—The default mapping.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>

## ieee-802.1ad

---

<b>Syntax</b>	ieee-802.1ad ( <i>rewrite-name</i>   default) <b>vlan-tag</b> (outer   outer-and-inner);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 9.2.
<b>Description</b>	Apply a IEEE-802.1ad rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules</b> <b>ieee-802.1ad</b>] hierarchy level.</p> <p><b>default</b>—The default rewrite bit mapping.</p> <p><b>vlan-tag</b>—The rewrite rule is applied to the <b>outer</b> or <b>outer-and-inner</b> VLAN tag.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li><li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li><li>• <a href="#">exp on page 377</a></li><li>• <a href="#">exp-push-push-push on page 487</a></li><li>• <a href="#">exp-swap-push-push on page 488</a></li><li>• <a href="#">ieee-802.1 (Rewrite Rules on Logical Interface) on page 379</a></li><li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## import (Rewrite Rules)

---

<b>Syntax</b>	<code>import (rewrite-name   default);</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service rewrite-rules type rewrite-name]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Specify a default or previously defined <b>rewrite-rules</b> mapping to import.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a rewrite-rules mapping configured at the <code>[edit class-of-service rewrite-rules]</code> hierarchy level.</p> <p><b>default</b>—The default <b>rewrite-rules</b> mapping.</p>
<b>Required Privilege Level</b>	<p><b>interface</b>—To view this statement in the configuration.</p> <p><b>interface-control</b>—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## inet-precedence (CoS Rewrite Rules)

---

<b>Syntax</b>	<code>inet-precedence (rewrite-name   default);</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Apply a IPv4 precedence rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the <code>[edit class-of-service rewrite-rules inet-precedence]</code> hierarchy level.</p> <p><b>default</b>—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.</p>
<b>Required Privilege Level</b>	<p><b>interface</b>—To view this statement in the configuration.</p> <p><b>interface-control</b>—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## inet-precedence (Rewrite Rules on Physical Interface)

---

<b>Syntax</b>	<code>inet-precedence (<i>rewrite-name</i>   default);</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.2 for the ACX Series Universal Access routers.
<b>Description</b>	Apply a IPv4 precedence rewrite rule.
<b>Options</b>	<p><b><i>rewrite-name</i></b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service rewrite-rules <b>inet-precedence</b>] hierarchy level.</p> <p><b>default</b>—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.</p>
<b>Required Privilege Level</b>	<p><b>interface</b>—To view this statement in the configuration.</p> <p><b>interface-control</b>—To add this statement to the configuration.</p>

## interfaces

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

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

**Hierarchy Level** [edit class-of-service]

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

**Description** Configure interface-specific CoS properties for incoming packets.


**Options** The remaining statements are explained separately.

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

**Related Documentation**

- [Overview of BA Classifier Types](#)
- [Configuring Rewrite Rules on page 215](#)

## loss-priority (BA Classifiers)

<b>Syntax</b>	<code>loss-priority <i>level</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service classifiers <i>type classifier-name</i> forwarding-class <i>class-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Specify packet loss priority value for a specific set of code-point aliases and bit patterns.
<b>Options</b>	<p><i>level</i> can be one of the following:</p> <ul style="list-style-type: none"> <li>• <b>high</b>—Packet has high loss priority.</li> <li>• <b>medium-high</b>—Packet has medium-high loss priority.</li> <li>• <b>medium-low</b>—Packet has medium-low loss priority.</li> <li>• <b>low</b>—Packet has low loss priority.</li> </ul>
<div>  <p><b>NOTE:</b> <b>medium-low</b> priority is not supported on PTX1000 routers.</p> </div>	
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26</a></li> <li>• <a href="#">Configuring Tricolor Marking on page 92</a></li> </ul>

## loss-priority-maps

---

<b>Syntax</b>	<pre>loss-priority-maps {     frame-relay-de <i>rewrite-name</i> {         loss-priority <i>level</i> {             <i>code-points</i> [ <i>aliases</i> ] [ <i>bit-patterns</i> ];         }     } }</pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced in JUNOS Release 11.4.
<b>Description</b>	Map the loss priority of incoming packets based on the CoS values.
<b>Options</b>	<p><b>frame-relay-de <i>rewrite-name</i></b>—Name of the Frame Relay DE bit loss priority map.</p> <p><b>loss-priority <i>level</i></b>—The loss priority level can be one of the following:</p> <ul style="list-style-type: none"><li>• <b>high</b>—Packet has high loss priority.</li><li>• <b>low</b>—Packet has low loss priority.</li><li>• <b>medium-high</b>—Packet has medium-high loss priority.</li><li>• <b>medium-low</b>—Packet has medium-low loss priority.</li></ul> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface</i></li></ul>

---

## loss-priority-maps (Assigning to an Interface)

---

<b>Syntax</b>	<code>loss-priority-maps {     frame-relay-de (<i>loss-priority-rewrite-name</i>   default); }</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i> ]
<b>Release Information</b>	Statement introduced in JUNOS Release 11.4.
<b>Description</b>	Assign the loss priority map to a logical interface.
<b>Options</b>	<p><b>default</b>—Apply the default loss priority map. The default map includes the following configuration:</p> <pre>loss-priority low code-point 0; loss-priority high code-point 1;</pre> <p><b>map-name</b>—Name of loss priority map to be applied.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Assigning the Default Frame Relay Discard Eligibility Loss Priority Map to an Interface</i></li><li>• <a href="#">unit on page 364</a></li></ul>

## protocol (Rewrite Rules)

<b>Syntax</b>	<code>protocol protocol-types;</code>
<b>Hierarchy Level</b>	<p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules exp <i>rewrite-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules dscp <i>rewrite-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules dscp-ipv6 <i>rewrite-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules inet-prec <i>rewrite-name</i>]</p>
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Option for <b>dscp</b> and <b>inet-prec</b> introduced in Junos OS Release 8.4.</p> <p>Option for <b>dscp-ipv6</b> introduced in Junos OS Release 10.4R2.</p>
<b>Description</b>	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 packets, and write the CoS value to MPLS and IPv4 headers.
<b>Options</b>	<p><b>protocol-types</b> can be one of the following:</p> <ul style="list-style-type: none"> <li>• <b>mpls</b>—Apply a rewrite rule to MPLS packets and write the CoS value to MPLS headers.</li> <li>• <b>mpls-inet-both</b>—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.</li> <li>• <b>mpls-inet-both-non-vpn</b>—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.</li> </ul>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Rewriting MPLS and IPv4 Packet Headers</i></li> </ul>

## rewrite-rules (CoS Host Outbound Traffic)

<b>Syntax</b>	rewrite-rules;
<b>Hierarchy Level</b>	[edit class-of-service host-outbound-traffic <a href="#">ieee-802.1</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 12.3.
<b>Description</b>	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.

<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 232</a></li> <li>• <a href="#">Rewriting Packet Headers to Ensure Forwarding Behavior on page 212</a></li> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## rewrite-rules (Definition)

---

<b>Syntax</b>	<pre>rewrite-rules {     type <i>rewrite-name</i>{         import (<i>rewrite-name</i>   default);         forwarding-class <i>class-name</i> {             loss-priority <i>level</i> <i>code-point</i> [ <i>aliases</i> ] [ <i>bit-patterns</i> ];         }     } }</pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. ieee-802.1ad option introduced in Junos OS Release 9.2.
<b>Description</b>	Specify a rewrite-rules mapping for the traffic that passes through all queues on the interface.
<b>Options</b>	<p><i>rewrite-name</i>—Name of a rewrite-rules mapping.</p> <p><i>type</i>—Traffic type.</p> <p><b>Values:</b> dscp, dscp-ipv6, exp, ieee-802.1, ieee-802.1ad, inet-precedence</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <i>Example: Configuring CoS for a PBB Network on MX Series Routers</i></li><li>• J Series router documentation</li></ul>

## rewrite-rules (Interfaces)

<b>Syntax</b>	<pre>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);   ieee-802.1ad (rewrite-name   default) vlan-tag (outer   outer-and-inner);   inet-precedence (rewrite-name   default); }</pre>
<b>Hierarchy Level</b>	<p>[edit class-of-service interfaces <i>interface-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i>]</p>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>Associate a rewrite-rules configuration or default mapping with a specific interface.</p> <p>The [edit class-of-service interfaces <i>interface-name</i>] hierarchy level is not supported on M Series routers.</p> <p>The [edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i>] hierarchy level is not supported on ACX Series routers.</p> <p>On MX Series routers, although you can configure firewall filters and CoS rewrite rules on IRB interfaces, we recommend that you do not configure these functionalities on IRB interfaces because they do not work properly.</p> <p>On an MX Series router and on an EX Series switch, <b>exp-push-push-push</b>, <b>exp-swap-push-push</b>, and <b>frame-relay-de</b> are not supported on an integrated routing and bridging (IRB) interface.</p> <p>On an ACX Series router, only the outer tag is supported for <b>dscp</b>, <b>inet-precedence</b>, and <b>ieee802.1</b>.</p> <p>On M Series routers only, if you include the <b>control-word</b> statement at the [edit protocols l2circuit neighbor address interface <i>interface-name</i>] hierarchy level, the software cannot rewrite MPLS EXP bits.</p> <p>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.</p> <p>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.</p>

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.

**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 215](#)
- [rewrite-rules \(Definition\) on page 504](#)
- [Applying Rewrite Rules to Output Logical Interfaces](#)

---

## 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 Access 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.

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

## unit

<b>Syntax</b>	<pre> unit logical-unit-number {   classifiers {     type (classifier-name   default) family (mpls   all);   }   forwarding-class class-name;   fragmentation-map map-name;   input-traffic-control-profile profiler-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; } </pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">interfaces</a> interface-name]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.
<b>Options</b>	<p><b>logical-unit-number</b>—Number of the logical unit.</p> <p><b>Range:</b> 0 through 16,384</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>Overview of BA Classifier Types</li> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## vlan-tag

---

<b>Syntax</b>	<code>vlan-tag (outer   outer-and-inner);</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules <a href="#">ieee-802.1</a> ( <i>rewrite-name</i>   default)]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.1.
<b>Description</b>	For Gigabit Ethernet IQ2 PICs only, apply this IEEE-802.1 rewrite rule to the outer or outer and inner VLAN tags.
<b>Default</b>	If you do not include this statement, the rewrite rule applies to the outer VLAN tag only.
<b>Options</b>	<b>outer</b> —Apply the rewrite rule to the outer VLAN tag only. <b>outer-and-inner</b> —Apply the rewrite rule to both the outer and inner VLAN tags.
<b>Required Privilege Level</b>	<b>interface</b> —To view this statement in the configuration. <b>interface-control</b> —To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Applying IEEE 802.1p Rewrite Rules to Dual VLAN Tags on page 216</a></li></ul>

## CHAPTER 28

# Configuration Statements: Routing Engine Protocol Queue Assignments

- [\[edit class-of-service\] Hierarchy Level](#) on page 509
- [classifiers \(Logical Interface\)](#) on page 514
- [dscp \(Rewrite Rules\)](#) on page 515
- [dscp-code-point \(CoS Host Outbound Traffic\)](#) on page 516
- [dscp-ipv6 \(CoS Rewrite Rules\)](#) on page 517
- [exp](#) on page 518
- [forwarding-class \(Forwarding Policy\)](#) on page 519
- [host-outbound-traffic \(Class-of-Service\)](#) on page 520
- [ieee-802.1 \(Rewrite Rules on Logical Interface\)](#) on page 521
- [inet-precedence \(CoS Rewrite Rules\)](#) on page 522
- [irb](#) on page 523
- [protocol \(Rewrite Rules\)](#) on page 524
- [rewrite-rules \(Interfaces\)](#) on page 525
- [unit](#) on page 527
- [vlan-tag](#) on page 528

### [\[edit class-of-service\] Hierarchy Level](#)

---

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

```
drop-profiles {
  profile-name {
    fill-level percentage drop-probability percentage;
    interpolate {
      drop-probability value;
      fill-level value;
    }
  }
}
fabric {
  scheduler-map {
    priority (high | low) scheduler scheduler-name;
  }
}
forwarding-class-map {
  map-name {
    class class-name queue-num queue-number <restricted-queue queue-number>;
  }
}
forwarding-classes {
  class class-name policing-priority (normal | premium) queue-num queue-number
  priority (high | low);
  queue queue-number class-name policing-priority (normal | premium) priority (high |
  low);
}
forwarding-policy {
  class class-name {
    classification-override {
      forwarding-class class-name;
    }
  }
  next-hop-map map-name {
    forwarding-class class-name {
      discard;
      lsp-next-hop [ lsp-regular-expressions ];
      next-hop [ next-hop-names ];
      non-lsp-next-hop;
    }
  }
}
fragmentation-maps {
  map-name {
    forwarding-class class-name {
      drop-timeout milliseconds;
      fragment-threshold bytes;
      multilink-class number;
      no-fragmentation;
    }
  }
}
host-outbound-traffic {
  dscp-code-point value;
  forwarding-class class-name;
  ieee-802.1 {
    default value;
    rewrite-rules;
```

```

    }
    tcp {
        raise-internet-control-priority;
    }
}
interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
}
}
restricted-queues {
    forwarding-class class-name queue-number;
}
rewrite-rules {
    (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
    rewrite-rule {
        forwarding-class class-name {
            loss-priority level code-point (alias | bits);
        }
        import (rewrite-rule | default);
    }
}
routing-instances routing-instance-name {
    classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
    }
}
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}
schedulers {
    scheduler-name {
        adjust-minimum value;
        adjust-percent value;
        buffer-size (exact | percent percentage | remainder);
        drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
            protocol any;
        excess-priority (high | low | medium-high | medium-low);
        excess-rate (percent percentage | proportion proportion);
        priority (high | low | medium-high | medium-low | strict-high);
        shaping-rate (bps | percent percentage | burst-size size);
        transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
    }
}
traceoptions {
    file <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
}
traffic-control-profiles {

```

```

    profile-name {
        adjust-minimum rate;
        delay-buffer-rate (bps | cps cps | percent percentage);
        excess-rate (percent percentage | proportion value);
        guaranteed-rate (bps | percent percentage) <burst-size bytes>;
        overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
        scheduler-map map-name;
        shaping-rate (bps | percent percentage) <burst-size bytes>;
    }
}
tri-color;
}

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

```

```

exp-push-push-push default;
exp-swap-push-push default;
ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
inet-precedence (rewrite-name | default) <protocol mpls>;
}
scheduler-map map-name;
shaping-rate bps;
translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
}
}
interface-set interface-set-name {
excess-bandwidth-share (equal | proportional value);
input-excess-bandwidth-share (equal | proportional value);
input-traffic-control-profile profile-name;
input-traffic-control-profile-remaining profile-name;
internal-node;
output-traffic-control-profile profile-name;
output-traffic-control-profile-remaining profile-name;
}
}
}


```

**Related Documentation**

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## classifiers (Logical Interface)

---


<b>Syntax</b>	<pre>classifiers {     type (classifier-name   default) family (mpls   inet); }</pre>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Apply a CoS aggregate behavior classifier to a logical interface. You can apply a default classifier or one that is previously defined.
<b>Options</b>	<p><b>classifier-name</b>—Name of the aggregate behavior classifier.</p> <p><b>type</b>—Traffic type.</p> <p><b>Values:</b> dscp, dscp-ipv6, exp, ieee-802.1, inet-precedence</p>
<hr/> <div> <b>NOTE:</b> You can only specify a family for the dscp and dscp-ipv6 types.</div> <hr/>	
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Default DSCP and DSCP IPv6 Classifiers on page 42</a></li><li>• <a href="#">Applying Behavior Aggregate Classifiers to Logical Interfaces on page 32</a></li></ul>

## dscp (Rewrite Rules)

<b>Syntax</b>	<code>dscp (rewrite-name   default) protocol mpls;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>For IPv4 traffic, apply a Differentiated Services (DiffServ) code point (DSCP) rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"> <li>On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li> <li>On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li> </ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules dscp</b>] hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> <li><a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li><a href="#">protocol (Rewrite Rules) on page 502</a></li> <li><a href="#">Rewriting MPLS and IPv4 Packet Headers</a></li> <li><a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## dscp-code-point (CoS Host Outbound Traffic)

---

<b>Syntax</b>	<code>dscp-code-point value;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">host-outbound-traffic</a> ]
<b>Release Information</b>	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.
<b>Description</b>	<p>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.</p> <p>If you use the <b>ping</b> operational mode command with the <b>tos type-of-service</b> option, the value specified in this configuration statement overrides the DSCP value you specify in the <b>ping</b> command.</p> <div> <b>NOTE:</b> Any DSCP rewrite rules configured on a 10-Gigabit Ethernet LAN/WAN PIC with SFP+ overwrite this DSCP value.</div> <p>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.</p>
<b>Options</b>	<b>code-point</b> —Six-bit DSCP code point value.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26</a></li><li>• <a href="#">Default DSCP and DSCP IPv6 Classifiers on page 42</a></li><li>• <a href="#">Changing the Default Queuing and Marking of Host Outbound Traffic on page 137.</a></li></ul>

## dscp-ipv6 (CoS Rewrite Rules)

<b>Syntax</b>	<code>dscp-ipv6 (rewrite-name   &lt;default&gt;) protocol mpls;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <a href="#">rewrite-rules</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Support for <b>protocol mpls</b> option introduced in Junos OS Release 10.4R2.
<b>Description</b>	<p>For IPv6 traffic, apply a DSCP rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp-ipv6</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"> <li>On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li> <li>On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li> </ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules dscp-ipv6</b>] hierarchy level.</p> <p><b>default</b>—Default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> <li><a href="#">protocol on page 502</a></li> <li><a href="#">Setting IPv6 DSCP and MPLS EXP Values Independently on page 217</a></li> <li><a href="#">Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 220</a></li> <li><a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li><a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## exp

---

<b>Syntax</b>	<code>exp (rewrite-name   default) protocol protocol-types;</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS Release 12.2. for ACX series
<b>Description</b>	Apply an MPLS experimental (EXP) rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the <code>[edit class-of-service rewrite-rules exp]</code> hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p>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 <b>mpls-inet-both</b> or <b>mpls-inet-both-non-vpn</b> option at the <code>[edit class-of-service interfaces interface interface-name unit logical-unit-number rewrite-rules exp rewrite-rule-name protocol]</code> hierarchy level. The IP precedence rewrite rules explanation does not apply to ACX Series Universal Access routers.</p> <p>On interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series 3D Universal Edge Routers and EX Series switches, we highly recommend that you configure the <b>default</b> option when you configure a behavior aggregate (BA) classifier that does not include a specific rewrite rule for MPLS packets. Doing so ensures that MPLS exp value is rewritten according to the BA classifier rules configured for forwarding or packet loss priority. This does not apply to ACX Series Universal Access routers.</p> <p><b>protocol-types</b>—Specify one or more protocol matching criteria:</p> <ul style="list-style-type: none"><li>• <b>mpls-any</b>—Apply to MPLS packets, write MPLS header only.</li><li>• <b>mpls-inet-both</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header.</li><li>• <b>mpls-inet-both-non-vpn</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header for only non VPN traffic.</li></ul>
<b>Required Privilege Level</b>	<b>interface</b> —To view this statement in the configuration. <b>interface-control</b> —To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</a></li><li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li><li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li></ul>

- [rewrite-rules \(Definition\)](#) on page 504

## forwarding-class (Forwarding Policy)

---

<b>Syntax</b>	<pre>forwarding-class <i>class-name</i> {   discard;   lsp-next-hop [ <i>lsp-regular-expression</i> ];   next-hop [ <i>next-hop-name</i> ];   non-lsp-next-hop; }</pre>
<b>Hierarchy Level</b>	[edit class-of-service forwarding-policy next-hop-map <i>map-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Define forwarding class name and associated next hops.
<b>Options</b>	<p><i>class-name</i>—Name of the forwarding class.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Overriding the Input Classification</a> on page 133</li><li>• <i>forwarding-class-default (Forwarding Policy)</i></li></ul>

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

---

<b>Syntax</b>	<pre>host-outbound-traffic {     forwarding-class <i>class-name</i>;     dscp-code-point <i>value</i>;     ieee-802.1 {         default <i>value</i>;         rewrite-rules;     } }</pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Statement introduced before Junos OS Release 11.4 for EX Series switches.</p> <p>Support for <b>ieee-802.1</b> statement introduced in Junos OS Release 12.3.</p> <p>Support for distributed protocol handler traffic introduced in Junos OS Release 13.2.</p>
<b>Description</b>	Classify and mark host outbound traffic. This statement does not affect transit traffic or incoming traffic.
<b>Default</b>	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.
<b>Options</b>	The remaining statements are explained separately.
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Default Queue Assignments for Routing Engine Sourced Traffic</a></li><li>• <a href="#">Default DSCP and DSCP IPv6 Classifiers on page 42</a></li><li>• <a href="#">Changing the Default Queuing and Marking of Host Outbound Traffic on page 137.</a></li><li>• <a href="#">Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic on page 232</a></li><li>• <a href="#">Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface on page 232</a></li></ul>

## ieee-802.1 (Rewrite Rules on Logical Interface)

<b>Syntax</b>	ieee-802.1 ( <i>rewrite-name</i>   default) <b>vlan-tag</b> (outer   outer-and-inner);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. <b>vlan-tag</b> statement introduced in Junos OS Release 8.1.
<b>Description</b>	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.
<b>Options</b>	<b>rewrite-name</b> —Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules</b> <b>ieee-802.1</b> ] hierarchy level.  <b>default</b> —The default mapping.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li> <li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li> <li>• <a href="#">exp on page 377</a></li> <li>• <a href="#">exp-push-push-push on page 487</a></li> <li>• <a href="#">exp-swap-push-push on page 488</a></li> <li>• <a href="#">ieee-802.1ad on page 381</a></li> <li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## inet-precedence (CoS Rewrite Rules)

---

<b>Syntax</b>	<code>inet-precedence (<i>rewrite-name</i>   default);</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <a href="#">rewrite-rules</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Apply a IPv4 precedence rewrite rule.
<b>Options</b>	<p><b><i>rewrite-name</i></b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service rewrite-rules inet-precedence] hierarchy level.</p> <p><b>default</b>—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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li><li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## 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 statements are explained separately.

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

**Related Documentation**

- *CoS Features and Limitations on MX Series Routers*

## protocol (Rewrite Rules)

<b>Syntax</b>	<code>protocol protocol-types;</code>
<b>Hierarchy Level</b>	<p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules exp <i>rewrite-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules dscp <i>rewrite-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules dscp-ipv6 <i>rewrite-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules inet-prec <i>rewrite-name</i>]</p>
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Option for <b>dscp</b> and <b>inet-prec</b> introduced in Junos OS Release 8.4.</p> <p>Option for <b>dscp-ipv6</b> introduced in Junos OS Release 10.4R2.</p>
<b>Description</b>	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 packets, and write the CoS value to MPLS and IPv4 headers.
<b>Options</b>	<p><b>protocol-types</b> can be one of the following:</p> <ul style="list-style-type: none"> <li>• <b>mpls</b>—Apply a rewrite rule to MPLS packets and write the CoS value to MPLS headers.</li> <li>• <b>mpls-inet-both</b>—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.</li> <li>• <b>mpls-inet-both-non-vpn</b>—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.</li> </ul>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Rewriting MPLS and IPv4 Packet Headers</i></li> </ul>

## rewrite-rules (Interfaces)

<b>Syntax</b>	<pre>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);   ieee-802.1ad (rewrite-name   default) vlan-tag (outer   outer-and-inner);   inet-precedence (rewrite-name   default); }</pre>
<b>Hierarchy Level</b>	<p>[edit class-of-service interfaces <i>interface-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i>]</p>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>Associate a rewrite-rules configuration or default mapping with a specific interface.</p> <p>The [edit class-of-service interfaces <i>interface-name</i>] hierarchy level is not supported on M Series routers.</p> <p>The [edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i>] hierarchy level is not supported on ACX Series routers.</p> <p>On MX Series routers, although you can configure firewall filters and CoS rewrite rules on IRB interfaces, we recommend that you do not configure these functionalities on IRB interfaces because they do not work properly.</p> <p>On an MX Series router and on an EX Series switch, <b>exp-push-push-push</b>, <b>exp-swap-push-push</b>, and <b>frame-relay-de</b> are not supported on an integrated routing and bridging (IRB) interface.</p> <p>On an ACX Series router, only the outer tag is supported for <b>dscp</b>, <b>inet-precedence</b>, and <b>ieee802.1</b>.</p> <p>On M Series routers only, if you include the <b>control-word</b> statement at the [edit protocols l2circuit neighbor address interface <i>interface-name</i>] hierarchy level, the software cannot rewrite MPLS EXP bits.</p> <p>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.</p> <p>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.</p>

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.

**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 215](#)
- [rewrite-rules \(Definition\) on page 504](#)
- *Applying Rewrite Rules to Output Logical Interfaces*

## unit

<b>Syntax</b>	<pre> unit <i>logical-unit-number</i> {   classifiers {     type (<i>classifier-name</i>   default) family (mpls   all);   }   forwarding-class <i>class-name</i>;   fragmentation-map <i>map-name</i>;   input-traffic-control-profile <i>profiler-name</i> shared-instance <i>instance-name</i>;   output-traffic-control-profile <i>profile-name</i> shared-instance <i>instance-name</i>;   per-session-scheduler;   rewrite-rules {     dscp (<i>rewrite-name</i>   default);     dscp-ipv6 (<i>rewrite-name</i>   default);     exp (<i>rewrite-name</i>   default) <i>protocol</i> <i>protocol-types</i>;     exp-push-push-push default;     exp-swap-push-push default;     ieee-802.1 (<i>rewrite-name</i>   default) <i>vlan-tag</i> (outer   outer-and-inner);     inet-precedence (<i>rewrite-name</i>   default);   }   scheduler-map <i>map-name</i>;   shaping-rate <i>rate</i>; } </pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">interfaces</a> <i>interface-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.
<b>Options</b>	<p><b><i>logical-unit-number</i></b>—Number of the logical unit.</p> <p><b>Range:</b> 0 through 16,384</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>Overview of BA Classifier Types</li> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## vlan-tag

---

<b>Syntax</b>	vlan-tag (outer   outer-and-inner);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules <a href="#">ieee-802.1</a> ( <i>rewrite-name</i>   default)]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.1.
<b>Description</b>	For Gigabit Ethernet IQ2 PICs only, apply this IEEE-802.1 rewrite rule to the outer or outer and inner VLAN tags.
<b>Default</b>	If you do not include this statement, the rewrite rule applies to the outer VLAN tag only.
<b>Options</b>	<b>outer</b> —Apply the rewrite rule to the outer VLAN tag only. <b>outer-and-inner</b> —Apply the rewrite rule to both the outer and inner VLAN tags.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Applying IEEE 802.1p Rewrite Rules to Dual VLAN Tags on page 216</a></li></ul>

# Configuration Statements: Schedulers

- [\[edit class-of-service\] Hierarchy Level](#) on page 530
- [buffer-size \(Schedulers\)](#) on page 534
- [delay-buffer-rate](#) on page 535
- [drop-profile-map \(Schedulers\)](#) on page 536
- [excess-priority](#) on page 537
- [excess-rate](#) on page 538
- [fabric \(Class-of-Service\)](#) on page 539
- [forwarding-class \(Interfaces\)](#) on page 539
- [guaranteed-rate](#) on page 540
- [interfaces](#) on page 541
- [loss-priority \(Scheduler Drop Profiles\)](#) on page 543
- [output-traffic-control-profile](#) on page 544
- [priority \(Fabric Queues, Schedulers\)](#) on page 545
- [priority \(Schedulers\)](#) on page 546
- [protocol \(Schedulers\)](#) on page 547
- [scheduler \(Fabric Queues\)](#) on page 548
- [scheduler \(Scheduler Map\)](#) on page 548
- [scheduler-map \(Fabric Queues\)](#) on page 549
- [scheduler-map \(Interfaces and Traffic-Control Profiles\)](#) on page 549
- [scheduler-map-chassis](#) on page 550
- [scheduler-maps \(For Most Interface Types\)](#) on page 551
- [schedulers \(CoS\)](#) on page 552
- [shaping-rate \(Applying to an Interface\)](#) on page 553
- [shaping-rate \(Oversubscribing an Interface\)](#) on page 555
- [traffic-control-profiles](#) on page 556
- [transmit-rate \(Schedulers\)](#) on page 557
- [unit](#) on page 559

- [\[edit interfaces\] Hierarchy Level](#) on page 559
- [schedulers \(Interfaces\)](#) on page 570

## [\[edit class-of-service\] Hierarchy Level](#)

---

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

```

        non-lsp-next-hop;
    }
}
}
fragmentation-maps {
    map-name {
        forwarding-class class-name {
            drop-timeout milliseconds;
            fragment-threshold bytes;
            multilink-class number;
            no-fragmentation;
        }
    }
}
host-outbound-traffic {
    dscp-code-point value;
    forwarding-class class-name;
    ieee-802.1 {
        default value;
        rewrite-rules;
    }
    tcp {
        raise-internet-control-priority;
    }
}
interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
}
}
restricted-queues {
    forwarding-class class-name queue-number;
}
rewrite-rules {
    (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
    rewrite-rule {
        forwarding-class class-name {
            loss-priority level code-point (alias | bits);
        }
        import (rewrite-rule | default);
    }
}
routing-instances routing-instance-name {
    classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
    }
}
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}
schedulers {

```

```

scheduler-name {
    adjust-minimum value;
    adjust-percent value;
    buffer-size (exact | percent percentage | remainder);
    drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
        protocol any;
    excess-priority (high | low | medium-high | medium-low);
    excess-rate (percent percentage | proportion proportion);
    priority (high | low | medium-high | medium-low | strict-high);
    shaping-rate (bps | percent percentage | burst-size size);
    transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
}
}
traceoptions {
    file <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
}
traffic-control-profiles {
    profile-name {
        adjust-minimum rate;
        delay-buffer-rate (bps | cps cps | percent percentage);
        excess-rate (percent percentage | proportion value);
        guaranteed-rate (bps | percent percentage) <burst-size bytes>;
        overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
        scheduler-map map-name;
        shaping-rate (bps | percent percentage) <burst-size bytes>;
    }
}
tri-color;
}

class-of-service {
    interfaces {
        interface-name {
            excess-bandwidth-share (equal | proportional value);
            input-excess-bandwidth-share (equal | proportional value);
            input-scheduler-map map-name;
            input-shaping-rate bps;
            input-traffic-control-profile profile-name;
            output-forwarding-class-map map-name;
            output-traffic-control-profile profile-name;
            scheduler-map map-name;
            scheduler-map-chassis (map-name | derived);
            shaping-rate bps;
            unit (logical-unit-number | *) {
                classifiers {
                    dscp (classifier-name | default) {
                        family [ inet mpls ];
                    }
                    dscp-ipv6 (classifier-name | default) {
                        family [ inet mpls ];
                    }
                    exp (classifier-name | default);
                    ieee-208.1 (classifier-name | default) <vlan-tag (inner | outer)>;
                }
            }
        }
    }
}

```

```

        ieee-208.1ad (classifier-name | default);
        inet-precedence (classifier-name | default);
    }
    forwarding-class class-name;
    input-scheduler-map map-name;
    input-shaping-rate bps;
    input-traffic-control-profile profile-name shared-instance instance-name;
    loss-priority-maps {
        (map-name | default);
    }
    loss-priority-rewrites {
        (map-name | default);
    }
    output-forwarding-class-map map-name;
    output-traffic-control-profile profile-name shared-instance instance-name;
    rewrite-rules {
        dscp (rule-name | default) <protocol mpls>;
        dscp-ipv6 (rule-name | default);
        exp (rule-name | default) <protocol [ mpls-any | mpls-inet-both |
            mpls-inet-both-non-vpn ]>;
        exp-push-push-push default;
        exp-swap-push-push default;
        ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        inet-precedence (rewrite-name | default) <protocol mpls>;
    }
    scheduler-map map-name;
    shaping-rate bps;
    translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
        to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
    }
}
interface-set interface-set-name {
    excess-bandwidth-share (equal | proportional value);
    input-excess-bandwidth-share (equal | proportional value);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
}
}
}


```

**Related Documentation**

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## buffer-size (Schedulers)

---

<b>Syntax</b>	buffer-size (percent <i>percentage</i>   remainder   temporal <i>microseconds</i> );
<b>Hierarchy Level</b>	[edit class-of-service schedulers <i>scheduler-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	Specify buffer size.
	<div> <b>NOTE:</b> On PTX Series Packet Transport Routers, buffer-size cannot be configured on rate-limited queues.</div>
<b>Default</b>	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.
<b>Options</b>	<p><b>percent <i>percentage</i></b>—Buffer size as a percentage of the total buffer. <b>Range:</b> 0 through 100</p> <p><b>remainder</b>—Remaining buffer available.</p> <p><b>temporal <i>microseconds</i></b>—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. <b>Range:</b> The ranges vary by platform as follows:</p> <ul style="list-style-type: none"><li>• For SRX Series Services Gateways: 1 through 2,000,000 microseconds.</li><li>• For vSRX instances: 1 through 32,000,000 microseconds.</li></ul>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Scheduler Buffer Size Overview</i></li></ul>

## delay-buffer-rate

<b>Syntax</b>	<code>delay-buffer-rate (percent <i>percentage</i>   <i>rate</i>);</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles</a> <i>profile-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 7.6.
<b>Description</b>	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.
<b>Default</b>	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 <a href="#">Table 32 on page 167</a> .
<b>Options</b>	<p><b>percent <i>percentage</i></b>—For LSQ interfaces, delay-buffer rate as a percentage of the available interface bandwidth.</p> <p><b>Range:</b> 1 through 100 percent</p> <p><b><i>rate</i></b>—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 <b>k</b> (1000), <b>m</b> (1,000,000), or <b>g</b> (1,000,000,000).</p> <p><b>Range:</b> 1000 through 6,400,000,000,000 bps</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Oversubscribing Interface Bandwidth on page 163</a></li> <li>• <a href="#">Providing a Guaranteed Minimum Rate on page 175</a></li> <li>• <a href="#">Configuring Traffic Control Profiles for Shared Scheduling and Shaping</a></li> <li>• <a href="#">output-traffic-control-profile on page 544</a></li> </ul>

## drop-profile-map (Schedulers)

---

<b>Syntax</b>	drop-profile-map <b>loss-priority</b> (any   low   medium-low   medium-high   high) <b>protocol</b> (any   non-tcp   tcp) <b>drop-profile (Schedulers)</b> <i>profile-name</i> ;
<b>Hierarchy Level</b>	[edit class-of-service <b>schedulers</b> <i>scheduler-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	Define the loss-priority value for a drop profile.  The statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Default Schedulers Overview on page 145</a></li><li>• <a href="#">Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199</a></li></ul>

## excess-priority


<b>Syntax</b>	<code>excess-priority [ low   medium-low   medium-high   high   none ];</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 9.3. Option <b>none</b> introduced in Junos OS Release 11.4.
<b>Description</b>	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.

<b>Options</b>	<p><b>low</b>—Excess traffic for this scheduler has low priority.</p> <p><b>medium-low</b>—Excess traffic for this scheduler has medium-low priority.</p> <p><b>medium-high</b>—Excess traffic for this scheduler has medium-high priority.</p> <p><b>high</b>—Excess traffic for this scheduler has high priority.</p> <p><b>none</b>—System does not demote the priority of guaranteed traffic when the bandwidth exceeds the shaping rate or the guaranteed rate.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Configuring Excess Bandwidth Sharing on IQE PICs</i></li> <li>• <a href="#">Bandwidth Sharing on Nonqueueing Packet Forwarding Engines Overview on page 179</a></li> <li>• <a href="#">Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 298</a></li> </ul>

## excess-rate

<b>Syntax</b>	<code>excess-rate (percent <i>percentage</i>   proportion <i>value</i>);</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ], [edit class-of-service <a href="#">traffic-control-profiles</a> <i>traffic-control-profile-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	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.
<div>  <b>NOTE:</b> The <b>proportion</b> option provides a greater range of values over the <b>percent</b> option and hence influences the priorities assigned to the queues. </div>	
<b>Options</b>	<p><b>percentage</b>—Percentage of the excess bandwidth to share.  <b>Range:</b> 0 through 100 percent  <b>Default:</b> Excess bandwidth is shared in proportion to the configured transmit rate of each queue.</p> <p><b>value</b>—(M Series, MX Series, T Series routers and EX Series switches only) Proportion of the excess bandwidth to share. Option available at the [edit class-of-service <a href="#">traffic-class-profiles</a> <i>traffic-control-profile-name</i>] hierarchy level only.  <b>Range:</b> 0 through 1000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Scheduler Transmission Rate on page 172</a></li> <li>• <a href="#">Configuring Excess Bandwidth Sharing on IQE PICs</a></li> <li>• <a href="#">Allocating Excess Bandwidth Among Frame Relay DLCIs on Multiservices PICs</a></li> <li>• <a href="#">Managing Excess Bandwidth Distribution on Static Interfaces on MICs and MPCs on page 298</a></li> </ul>

## fabric (Class-of-Service)

<b>Syntax</b>	<pre>fabric {     scheduler-map {         priority (high   low) scheduler scheduler-name;     } }</pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Statement introduced before Junos OS 11.4 for EX Series switches.</p>
<b>Description</b>	<p>Define CoS parameters of the switch fabric. For M320 and T Series routers only, associate a scheduler with a fabric priority.</p> <p>On EX Series switches, this statement is supported only on EX8200 standalone switches and EX8200 Virtual Chassis.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>See <a href="#">Associating Schedulers with Fabric Priorities on page 197</a>.</li> </ul>

## forwarding-class (Interfaces)

<b>Syntax</b>	forwarding-class <i>class-name</i> ;
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> ]
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Statement introduced in Junos OS Release 12.2 for ACX Series routers.</p>
<b>Description</b>	Associate a forwarding class configuration or default mapping with a specific interface.
<b>Options</b>	<i>class-name</i> —Name of the forwarding class.
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li><a href="#">Applying Forwarding Classes to Interfaces on page 134</a></li> </ul>

## guaranteed-rate

---

<b>Syntax</b>	<code>guaranteed-rate (percent <i>percentage</i>   <i>rate</i>) &lt;burst-size <i>bytes</i>&gt;;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles</a> <i>profile-name</i> ]
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 7.6.</p> <p>Option <b>burst-size</b> introduced for Enhanced Queuing (EQ) DPC interfaces in Junos OS Release 9.4.</p> <p>Option <b>burst-size</b> introduced for MIC and MPC interfaces in Junos OS Release 11.4.</p> <p>Option <b>burst-size</b> introduced for IQ2 and IQ2E interfaces in Junos OS Release 12.3</p>
<b>Description</b>	For Gigabit Ethernet IQ, Channelized IQ PICs, AS PIC 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.
<b>Default</b>	If you do not include this statement and you do not include the <b>delay-buffer-rate</b> statement, the logical interface receives a minimal delay-buffer rate and minimal bandwidth equal to 2 MTU-sized packets.
<b>Options</b>	<p><b>percent <i>percentage</i></b>—For LSQ interfaces, guaranteed rate as a percentage of the available interface bandwidth.</p> <p><b>Range:</b> 1 through 100 percent</p> <p><b><i>rate</i></b>—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 <b>k</b> (1000), <b>m</b> (1,000,000), or <b>g</b> (1,000,000,000).</p> <p><b>Range:</b> 1000 through 6,400,000,000,000 bps</p> <p><b>burst-size <i>bytes</i></b>—(Optional) Maximum burst size, in bytes.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Providing a Guaranteed Minimum Rate on page 175</a></li><li>• <a href="#">Configuring Traffic Control Profiles for Shared Scheduling and Shaping</a></li><li>• <a href="#">output-traffic-control-profile on page 544</a></li></ul>

## interfaces

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

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

**Hierarchy Level** [edit class-of-service]

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

**Description** Configure interface-specific CoS properties for incoming packets.

**Options** The remaining statements are explained separately.

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

**Related Documentation**

- *Overview of BA Classifier Types*
- [Configuring Rewrite Rules on page 215](#)

## loss-priority (Scheduler Drop Profiles)

<b>Syntax</b>	loss-priority (any   high   low   medium-high   medium-low);
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> drop-profile-map]
<b>Release Information</b>	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.
<b>Description</b>	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.
<b>Options</b>	<b>any</b> —The drop profile applies to packets with any PLP.



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

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

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


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

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

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

<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Default Schedulers Overview on page 145</a></li> <li>• <a href="#">Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199</a></li> <li>• <a href="#">Configuring Schedulers for Priority Scheduling on page 195</a></li> <li>• <a href="#">Configuring Tricolor Marking on page 92</a></li> <li>• <a href="#">protocol (Schedulers) on page 359</a></li> </ul>
------------------------------	---

## output-traffic-control-profile

<b>Syntax</b>	<code>output-traffic-control-profile <i>profile-name</i> shared-instance <i>instance-name</i>;</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces <i>interface-name</i> ],</code> <code>[edit class-of-service interfaces <i>interface-name</i> <i>unit</i> <i>logical-unit-number</i>],</code> <code>[edit class-of-service interfaces <i>interface-name</i> interface-set <i>interface-set-name</i>]</code>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 7.6.</p> <p><b>interface-set</b> option added for Enhanced Queuing DPCs on MX Series routers in Junos OS Release 8.5.</p> <p><b>interface-set</b> option added for MIC and MPC interfaces on MX Series routers in Junos OS Release 10.2.</p> <p>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.</p>
<b>Description</b>	<p>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.</p> <p>The statement is supported on the following interfaces:</p> <ul style="list-style-type: none"> <li>• Channelized IQ PIC interfaces</li> <li>• Gigabit Ethernet IQ, Gigabit Ethernet IQ2, and IQ2E PIC interfaces</li> <li>• Link services IQ (LSQ) interfaces on AS PICs</li> <li>• Enhanced Queuing DPC, MIC, and MPC interfaces on MX Series routers</li> <li>• GRE tunnel interfaces configured on physical or logical interfaces hosted on MIC or MPC line cards in MX Series routers.</li> </ul>
	<p> <b>NOTE:</b> Interface sets (sets of interfaces used to configure hierarchical CoS schedulers on supported Ethernet interfaces) are not supported on GRE tunnel interfaces.</p>
	<p>The <b>shared-instance</b> statement is supported on Gigabit Ethernet IQ2 PICs only.</p>
<b>Options</b>	<b><i>profile-name</i></b> —Name of the traffic-control profile to be applied to this interface
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Oversubscribing Interface Bandwidth on page 163</a></li> <li>• <i>Configuring Traffic Control Profiles for Shared Scheduling and Shaping</i></li> <li>• <i>Configuring Hierarchical Schedulers for CoS</i> (Enhanced Queuing DPC, MIC, and MPC interfaces on MX Series routers)</li> </ul>

- *Configuring Interface Sets* (Enhanced Queuing DPC, MIC, and MPC interfaces on MX Series routers)
- *output-traffic-control-profile-remaining*
- [traffic-control-profiles on page 432](#)

---


## priority (Fabric Queues, Schedulers)

---

<b>Syntax</b>	<code>priority (high   low)scheduler scheduler-name;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">fabric scheduler-map</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS 11.4 for EX Series switches.
<b>Description</b>	<p>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.</p> <p>For a scheduler that you associate with a fabric priority, you cannot include the <b>buffer-size</b>, <b>transmit-rate</b>, or <b>priority</b> statements at the [edit class-of-service schedulers <b>scheduler-name</b>] hierarchy level.</p> <p>On EX Series switches, this statement is supported only on EX8200 standalone switches and EX8200 Virtual Chassis.</p>
<b>Options</b>	<p><b>high</b>—Scheduler has high priority.</p> <p><b>low</b>—Scheduler has low priority.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• See <a href="#">Associating Schedulers with Fabric Priorities on page 197</a>.</li></ul>

## priority (Schedulers)

---

<b>Syntax</b>	<code>priority <i>priority-level</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	Specify the packet-scheduling priority value.
<b>Options</b>	<p><i>priority-level</i> can be one of the following:</p> <ul style="list-style-type: none"><li>• <b>low</b>—Scheduler has low priority.</li><li>• <b>medium-low</b>—Scheduler has medium-low priority.</li><li>• <b>medium-high</b>—Scheduler has medium-high priority.</li><li>• <b>high</b>—Scheduler has high priority. Assigning high priority to a queue prevents the queue from being underserved.</li><li>• <b>strict-high</b>—Scheduler has strictly high priority. Configure a <b>high</b> priority queue with unlimited transmission bandwidth available to it. As long as it has traffic to send, the <b>strict-high</b> priority queue receives precedence over <b>low</b>, <b>medium-low</b>, and <b>medium-high</b> priority queues, but not <b>high</b> priority queues. You can configure <b>strict-high</b> priority on only one queue per interface.</li></ul>
<div> <b>NOTE:</b> The <b>strict-high</b> 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.</div>	
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Schedulers for Priority Scheduling on page 195</a></li></ul>

## protocol (Schedulers)

---

<b>Syntax</b>	protocol (any   non-tcp   tcp);
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> <a href="#">drop-profile-map</a> ]
<b>Release Information</b>	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.
<b>Description</b>	Specify the protocol type for the specified scheduler.
<b>Options</b>	<b>any</b> —Accept any protocol type.  <b>non-tcp</b> —(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](#).

---

	<b>tcp</b> —(ACX Series Routers, M Series and T Series (except T4000) routers only) Accept TCP/IP protocol type.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Schedulers on page 146</a></li></ul>

## scheduler (Fabric Queues)

---

<b>Syntax</b>	<code>scheduler <i>scheduler-name</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">fabric scheduler-map</a> <i>priority</i> (high   low)]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS Release 11.4 for EX Series switches.
<b>Description</b>	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.
<b>Options</b>	<i>scheduler-name</i> —Name of the scheduler configuration block.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>See <a href="#">Associating Schedulers with Fabric Priorities on page 197</a>.</li></ul>

## scheduler (Scheduler Map)

---

<b>Syntax</b>	<code>scheduler <i>scheduler-name</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service scheduler-maps <i>map-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 12.2 for ACX Series Routers.
<b>Description</b>	Associate a scheduler with a scheduler map.
<b>Options</b>	<i>scheduler-name</i> —Name of the scheduler configuration block.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li><a href="#">Configuring Schedulers on page 146</a></li></ul>

## scheduler-map (Fabric Queues)

---

<b>Syntax</b>	<code>scheduler-map</code> <b>priority</b> (high   low) <b>scheduler</b> <i>scheduler-name</i> ;
<b>Hierarchy Level</b>	[edit class-of-service <b>fabric</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS Release 11.4 for EX Series switches.
<b>Description</b>	For M320, MX Series, and T Series routers only, associate a scheduler with a fabric priority.  The statements are explained separately.
<b>Usage Guidelines</b>	See “ <a href="#">Associating Schedulers with Fabric Priorities</a> ” on page 197.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• See <a href="#">Associating Schedulers with Fabric Priorities</a> on page 197.</li></ul>

## scheduler-map (Interfaces and Traffic-Control Profiles)

---

<b>Syntax</b>	<code>scheduler-map</code> <i>map-name</i> ;
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> ], [edit class-of-service interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i> ], [edit class-of-service <b>traffic-control-profiles</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	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.
<b>Options</b>	<b>map-name</b> —Name of the scheduler map.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Schedulers</a> on page 146</li><li>• <a href="#">Oversubscribing Interface Bandwidth</a> on page 163</li><li>• <a href="#">output-traffic-control-profile</a> on page 544</li></ul>

## scheduler-map-chassis

---

<b>Syntax</b>	<code>scheduler-map-chassis (derived   <i>map-name</i>);</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-type-fpc/pic/*</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	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.
<b>Default</b>	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.
<b>Options</b>	<b>derived</b> —Sets the chassis queues to derive their scheduling configuration from the associated logical interface scheduling configuration.  <b><i>map-name</i></b> —Name of the scheduler map configured at the [edit class-of-service scheduler-maps] hierarchy level.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Applying Scheduler Maps to Packet Forwarding Component Queues</i></li><li>• <a href="#">scheduler-map (Fabric Queues) on page 549</a></li></ul>

## **scheduler-maps (For Most Interface Types)**

---


<b>Syntax</b>	<pre>scheduler-maps {   map-name {     forwarding-class class-name scheduler scheduler-name;   } }</pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Specify a scheduler map name and associate it with the scheduler configuration and forwarding class.
<b>Options</b>	<p><i>map-name</i>—Name of the scheduler map.</p> <p>The remaining statements are explained separately.</p> <p>See <a href="#">“Configuring Schedulers” on page 146</a> .</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>

## schedulers (CoS)

---

Syntax	<pre>schedulers {   scheduler-name {     adjust-minimum <i>rate</i>;     adjust-percent <i>percentage</i>;     buffer-size (<i>seconds</i>   percent <i>percentage</i>   remainder   temporal <i>microseconds</i>);     drop-profile-map <i>loss-priority</i> (any   low   medium-low   medium-high   high) <i>protocol</i>       (any   non-tcp   tcp) <i>drop-profile</i> <i>profile-name</i>;     excess-priority [ low   medium-low   medium-high   high   none];     excess-rate (percent <i>percentage</i>   proportion <i>value</i>);     priority <i>priority-level</i>;     shaping-rate (percent <i>percentage</i>   <i>rate</i>);     transmit-rate (percent <i>percentage</i>   <i>rate</i>   remainder) &lt;exact   rate-limit&gt;;   } }</pre>
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	<i>scheduler-name</i> —Name of the scheduler to be configured.  The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"><li>• <a href="#">Schedulers Overview on page 143</a></li><li>• <a href="#">Default Schedulers Overview on page 145</a></li><li>• <a href="#">Configuring Schedulers on page 146</a></li><li>• <a href="#">Configuring a Scheduler</a></li></ul>

## shaping-rate (Applying to an Interface)

<b>Syntax</b>	<code>shaping-rate rate;</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces <i>interface-name</i>],</code> <code>[edit class-of-service interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i>]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. <code>[edit class-of-service interfaces <i>interface-name</i>]</code> hierarchy level added in Junos OS Release 7.5. Statement introduced in Junos OS Release 13.2 on PTX Series Packet Transport Routers.
<b>Description</b>	<p>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.</p> <p>For physical interfaces on IQ PICs and T4000 routers with Type 5 FPCs only, configure traffic shaping based on the rate-limited bandwidth of the total interface bandwidth.</p> <p>Logical and physical interface traffic shaping rates are mutually exclusive. This means you can include the <b>shaping-rate</b> statement at the <code>[edit class-of-service interfaces <i>interface-name</i>]</code> hierarchy level or the <code>[edit class-of-service interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i>]</code> hierarchy level, but not both.</p>
	<p> <b>NOTE:</b> For MX Series routers and for EX Series switches, the shaping rate value for the physical interface at the <code>[edit class-of-service interfaces <i>interface-name</i>]</code> hierarchy level must be a minimum of 160 Kbps. If the value is less than the sum of the logical interface guaranteed rates, the user is not allowed to apply the shaping rate to a physical interface.</p> <p>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 <b>shaping-rate</b> is limited by the maximum transmission rate of the interface.</p>
	<p>Alternatively, you can configure a shaping rate for a logical interface and oversubscribe the physical interface by including the <b>shaping-rate</b> statement at the <code>[edit class-of-service <b>traffic-control-profiles</b>]</code> hierarchy level. With this configuration approach, you can independently control the delay-buffer rate, as described in <a href="#">“Oversubscribing Interface Bandwidth” on page 163</a>.</p> <p>For FRF.15 and FRF.16 bundles on link services interfaces, only shaping rates based on percentage are supported.</p>
<b>Default</b>	If you do not include this statement at the <code>[edit class-of-service interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i>]</code> 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 <code>[edit class-of-service interfaces <i>interface-name</i>]</code> 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:** For all MX Series and EX series interfaces, the rate can be from 65,535 through 6,400,000,000,000 bps.

---

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

**Related Documentation**

- [Applying Scheduler Maps Overview on page 159](#)
- *Configuring Virtual LAN Queuing and Shaping on PTX Series Routers*

## shaping-rate (Oversubscribing an Interface)

<b>Syntax</b>	<code>shaping-rate (percent <i>percentage</i>   <i>rate</i>) &lt;burst-size <i>bytes</i>&gt;;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">traffic-control-profiles</a> <i>profile-name</i> ]
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 7.6.</p> <p>Option <b>burst-size</b> introduced for Enhanced Queuing (EQ) DPC interfaces on MX Series routers in Junos OS Release 9.4.</p> <p>Option <b>burst-size</b> option introduced for MIC and MPC interfaces on MX Series routers in Junos OS Release 11.4.</p> <p>Option <b>burst-size</b> introduced for IQ2 and IQ2E interfaces in Junos OS Release 12.3.</p>
<b>Description</b>	<p>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.</p> <p>For physical interfaces on T4000 router interfaces on Type 5 FPCs, configure traffic shaping rate.</p> <p>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).</p>
<b>Default</b>	The default behavior depends on various factors. For more information, see <a href="#">Table 32 on page 167</a> .
<b>Options</b>	<p><b>percent <i>percentage</i></b>—For LSQ interfaces, shaping rate as a percentage of the available interface bandwidth.</p> <p><b>Range:</b> 1 through 100 percent</p> <p><b><i>rate</i></b>—For IQ and IQ2 interfaces, and T4000 routers with Type 5 FPCs, 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 <b>k</b> (1000), <b>m</b> (1,000,000), or <b>g</b> (1,000,000,000).</p> <p><b>Range:</b> IQ and IQ2 interfaces—1000 through 6,400,000,000,000 bps</p> <p><b>Range:</b> 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 <b>shaping-rate</b> is limited by the maximum transmission rate of the interface.</p> <p><b>burst-size <i>bytes</i></b>—(Optional) Maximum burst size, in bytes.</p> <p><b>Range:</b> 0 through 1,000,000,000</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>

- Related Documentation**
- [Configuring Traffic Control Profiles for Shared Scheduling and Shaping](#)
  - [Oversubscribing Interface Bandwidth on page 163](#)
  - [output-traffic-control-profile on page 544](#)

## traffic-control-profiles

<b>Syntax</b>	<pre> traffic-control-profiles <i>profile-name</i> {   adjust-minimum <i>rate</i>;   atm-service (cbr   rtvbr   nrtvbr);   delay-buffer-rate (percent <i>percentage</i>   <i>rate</i>);   excess-rate (percent <i>percentage</i>   proportion <i>value</i> );   excess-rate-high (percent <i>percentage</i>   proportion <i>value</i>);   excess-rate-low (percent <i>percentage</i>   proportion <i>value</i>);   guaranteed-rate (percent <i>percentage</i>   <i>rate</i>) &lt;burst-size <i>bytes</i>&gt;;   max-burst-size <i>cells</i>;   overhead-accounting (frame-mode   cell-mode   frame-mode-bytes   cell-mode-bytes)     &lt;bytes (<i>byte-value</i>)&gt;;   peak-rate <i>rate</i>;   scheduler-map <i>map-name</i>;   shaping-rate (percent <i>percentage</i>   <i>rate</i>) &lt;burst-size <i>bytes</i>&gt;;   shaping-rate-excess-high <i>rate</i> [ burst-size <i>bytes</i> ];   shaping-rate-excess-low <i>rate</i> [ burst-size <i>bytes</i> ];   shaping-rate-priority-high <i>rate</i> [ burst-size <i>bytes</i> ];   shaping-rate-priority-low <i>rate</i> [ burst-size <i>bytes</i> ];   shaping-rate-priority-medium <i>rate</i> [ burst-size <i>bytes</i> ];   strict-priority-scheduler;   sustained-rate <i>rate</i>; } </pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced in Junos OS Release 7.6.
<b>Description</b>	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 <b>excess-rate</b> statement.
<b>Options</b>	<p><b><i>profile-name</i></b>—Name of the traffic-control profile.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Oversubscribing Interface Bandwidth on page 163</a></li> <li>• <a href="#">Understanding Scheduling on PTX Series Routers</a></li> <li>• <a href="#">output-traffic-control-profile on page 544</a></li> </ul>

## transmit-rate (Schedulers)

<b>Syntax</b>	<code>transmit-rate (rate   percent <i>percentage</i>   remainder) &lt;exact   rate-limit&gt;;</code>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> ]
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p><b>rate-limit</b> option introduced in Junos OS Release 8.3. Applied to the Multiservices PICs in Junos OS Release 9.4.</p> <p>Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Routers.</p> <p>Statement introduced in Junos OS Release 12.2 for ACX Series Routers.</p>
<b>Description</b>	Specify the transmit rate or percentage for a scheduler.
<b>Default</b>	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.
<b>Options</b>	<p><b>exact</b>—(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 Packet Transport Routers, this option is allowed only on the non-strict-high (high, medium-high, medium-low, or low) queues.</p> <p><b>percent <i>percentage</i></b>—Percentage of transmission capacity. A percentage of zero drops all packets in the queue.</p> <p><b>Range:</b> 0 through 100 percent for M, MX and T Series routers and EX Series switches; 1 through 100 percent for PTX Series Packet Transport Routers; 0 through 200 percent for the SONET/SDH OC48/STM16 IQE PIC</p>



### 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 `schedulers` *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.

**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 Packet Transport 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 146](#)
- [Configuring Scheduler Transmission Rate on page 172](#)
- [Understanding Scheduling on PTX Series Routers](#)

## unit

<b>Syntax</b>	<pre> unit logical-unit-number {   classifiers {     type (classifier-name   default) family (mpls   all);   }   forwarding-class class-name;   fragmentation-map map-name;   input-traffic-control-profile profiler-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; } </pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">interfaces</a> interface-name]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.
<b>Options</b>	<p><b>logical-unit-number</b>—Number of the logical unit.</p> <p><b>Range:</b> 0 through 16,384</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>Overview of BA Classifier Types</li> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## [edit interfaces] Hierarchy Level

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

```

interfaces {
  interface-name {
    ... the "interface-name" subhierarchy appears after the main [edit interfaces] hierarchy level ...
  }
}

```

```
}
interface-set interface-set-name {
  interface interface-name {
    (unit unit-number | vlan-tags-outer vlan-tag);
  }
}
irb {
  accounting-profile name;
  description text;

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

  traceoptions {
    flag flag;
  }
  (traps | no-traps);
  unit logical-unit-number {
    accounting-profile name;
    bandwidth rate;
    description text;
    disable;
    encapsulation type;
    family inet {
      accounting {
        destination-class-usage;
        source-class-usage {
          input;
          output;
        }
      }
    }
  }
  address ipv4-address {
    arp ip-address (mac | multicast-mac) mac-address <publish>;
    broadcast address;
    preferred;
    primary;
    vrrp-group group-id {
      (accept-data | no-accept-data);
      advertise-interval seconds;
      advertisements-threshold number;
      authentication-key key;
      authentication-type authentication;
      fast-interval milliseconds;
      (preempt | no-preempt) {
        hold-time seconds;
      }
    }
    priority number;
    track {
      interface interface-name {
        bandwidth-threshold bits-per-second priority-cost priority;
        priority-cost priority;
      }
    }
    priority-hold-time seconds;
  }
}
```

```

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

```

```

    }
    virtual-inet6-address [addresses];
    virtual-link-local-address ipv6-address;
    vrrp-inherit-from {
        active-group group-number;
        active-interface interface-name;
    }
}
}
(dad-disable | no-dad-disable);
filter {
    input filter-name;
    output filter-name;
}
mtu bytes;
nd6-stale-time seconds;
no-neighbor-learn;
no-redirects;
policer {
    input policer-name;
    output policer-name;
}
rpf-check {
    fail-filter filter-name;
    mode {
        loose;
    }
}
}
}
family iso {
    address interface-address;
    mtu bytes;
}
family mpls {
    filter {
        input filter-name;
        output filter-name;
    }
    mtu bytes;
    policer {
        input policer-name;
        output policer-name;
    }
}
native-inner-vlan-id vlan-id;
proxy-arp (restricted | unrestricted);
(traps | no-traps);
vlan-id-list [vlan-id's];
vlan-id-range [vlan-id-range];
}
}
traceoptions {
    file <filename> <files number> <match regular-expression> <size maximum-file-size>
    <world-readable | no-world-readable>;
    flag flag <disable>;
    no-remote-trace;
}

```

```

    }
}

interfaces {
  interface-name {
    disable;
    accounting-profile name;
    aggregated-ether-options {
      ethernet-switch-profile {
        tag-protocol-id [ hexadecimal-identifiers ];
      }
      (flow-control | no-flow-control);
      lacp {
        (active | passive);
        admin-key key;
        fast-failover;
        link-protection {
          disable;
          (revertive | non-revertive);
        }
        periodic (fast | slow);
        system-id mac-address;
        system-priority priority;
      }
      (link-protection | no-link-protection);
      link-speed (100m | 1g | 8g | 10g | 40g | 50g | 80g | 100g | oc192);
      logical-interface-fpc-redundancy;
      (loopback | no-loopback);
      mc-ae {
        chassis-id chassis-id;
        events {
          iccp-peer-down {
            force-icl-down;
            prefer-status-control-active;
          }
        }
        mc-ae-id mc-ae-id;
        mode (active-active | active-standby);
        redundancy-group group-id;
        status-control (active | standby);
      }
      minimum-links number;
      rebalance-periodic {
        start-time time;
        interval number;
      }
      source-address-filter {
        mac-address;
      }
      (source-filtering | no-source-filtering);
    }
  }
  auto-configure {
    remove-when-no-subscribers;
    stacked-vlan-ranges {
      access-profile profile-name;
      authentication {

```

```

    password password-string;
    username-include {
        circuit-type;
        delimiter delimiter-character;
        domain-name domain-name-string;
        interface-name;
        mac-address;
        option-82 ( circuit-id | remote-id);
        radius-realm radius-realm-string;
        user-prefix user-prefix-string;
    }
}
dynamic-profile profile-name {
    accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
    ranges (any | low-tag-high-tag), (any | low-tag-high-tag);
}
}
vlan-ranges {
    access-profile profile-name;
    authentication {
        password password-string;
        username-include {
            circuit-type;
            delimiter delimiter-character;
            domain-name domain-name-string;
            interface-name;
            mac-address;
            option-82;
            radius-realm radius-realm-string;
            user-prefix user-prefix-string;
        }
    }
}
dynamic-profile profile-name {
    accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
    ranges (any | low-tag)—(any | high-tag);
}
}
override tag vlan-tag dynamic-profile profile name;
}
encapsulation (ethernet-bridge | ethernet-vpls | extended-vlan-bridge |
    extended-vlan-vpls | flexible-ethernet-services | vlan-vpls);
ether-options {
    802.3ad {
        aex;
        (backup | primary);
        lacp {
            force-up;
            port-priority
        }
    }
}
asynchronous-notification;
(auto-negotiation | no-auto-negotiation);
ethernet-switch-profile {
    ethernet-policer-profile {
        input-priority-map {
            ieee802.1p premium [ values ];

```

```

    }
    output-priority-map {
        classifier {
            premium {
                forwarding-class class-name {
                    loss-priority (high | low);
                }
            }
        }
    }
    policer cos-policer-name {
        aggregate {
            bandwidth-limit bps;
            burst-size-limit bytes;
        }
        premium {
            bandwidth-limit bps;
            burst-size-limit bytes;
        }
    }
    tag-protocol-id;
}
(mac-learn-enable | no-mac-learn-enable);
}
(flow-control | no-flow-control);
ignore-l3-incompletes;
link-mode (automatic | full-duplex | half-duplex);
(lloopback | no-loopback);
keepalives <interval seconds> <down-count number> <up-count number>;
speed (1g | 10m | 100m | 10m-100m | auto-negotiation);
source-address-filter {
    mac-address;
}
source-filtering | no-source-filtering;
}
flexible-vlan-tagging;
(gratuitous-arp-reply | no-gratuitous-arp-reply);
hold-time (up milliseconds | down milliseconds);
interface-transmit-statistics;
(keepalives <down-count number> <interval seconds> <up-count number> |
no-keepalives);
layer2-policer {
    apply-groups [ group-names ];
    apply-groups-except [ group-names ];
}
link-mode (automatic | full-duplex);
mac mac-address;
mtu bytes;
multi-chassis-protection peer-ip-address {
    interface interface-name;
}
native-vlan-id number;
no-gratuitous-arp-request;
optics-options {
    alarm low-light-alarm {
        (link-down | syslog);
    }
}

```

```

    }
    warning low-light-warning {
        (link-down | syslog);
    }
    wavelength nm;
}
passive-monitor-mode;
per-unit-scheduler;
speed (10m | 100m | 1g | auto | oc3 | oc12 | oc48);
stacked-vlan-tagging;
traceoptions {
    flag flag;
}
transmit-bucket {
    overflow discard;
    rate percentage;
    threshold bytes;
}
(traps | no-traps);
unidirectional;
vlan-tagging;
}

```

```

interface-name {
    unit logical-unit-number {
        disable;
        accept-source-mac {
            mac-address mac-address {
                policer {
                    input policer-name;
                    output policer-name;
                }
            }
        }
        accounting-profile name;
        advisory-options {
            downstream-rate rate;
            upstream-rate rate;
        }
        arp-resp (restricted|unrestricted);
        bandwidth rate;
        clear-dont-fragment-bit;
        copy-tos-to-outer-ip-header;
        demux-destination family;
        encapsulation (vlan-bridge | vlan-vpls);
        epd-threshold cells plp1 cells;
        filter filter-name;
        inner-vlan-id-range start start-id end end-id;
        input-vlan-map {
            (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
            inner-tag-protocol-id tpid;
            inner-vlan-id number;
            tag-protocol-id tpid;
            vlan-id number;
        }
    }
}

```

```

interface-shared-with psdnumerical-index;
layer2-policer {
    input-hierarchical-policer policer-name;
    input-policer policer-name;
    input-three-color policer-name;
    output-policer policer-name;
    output-three-color policer-name;
}
multi-chassis-protection peer-ip-address {
    interface interface-name;
}
native-inner-vlan-id number;
output-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
    inner-tag-protocol-id tpid;
    inner-vlan-id number;
    tag-protocol-id tpid;
    vlan-id number;
}
peer-interface interface-name;
peer-unit unit-number;
plp-to-clp;
proxy-arp <restricted | unrestricted>;
rpm {
    (client | server);
    twamp-server;
}
swap-by-poppush;
vlan-id number;
vlan-id-list [ vlan-id vlan-id-vlan-id ];
vlan-id-range number-number;
vlan-tags (inner <tpid.>vlan-id | inner-list [ vlan-id vlan-id-vlan-id ] |
    inner-range <tpid.>vlan-id-vlan-id) outer <tpid.>vlan-id;
}

unit logical-unit-number {
    family ethernet-switching {
        filter {
            group filter-group-number;
            (input filter-name | input-list [ filter-names ]);
            (output filter-name | output-list [ filter-names ]);
            (inner-vlan-id-list [ vlan-ids ] | vlan-id number | vlan-id-list [ number
                number-number ]);
            interface-mode (access | trunk);
            policer {
                input policer-name;
                output policer-name;
            }
            vlan-rewrite {
                translate old-vlan-id new-vlan-id;
            }
            vlan {
                members [ all vlan-identifiers ];
            }
        }
    }
    family inet {

```

```

filter {
    group filter-group-number;
    (input filter-name | input-list [ filter-names ]);
    (output filter-name | output-list [ filter-names ]);
}
input-hierarchical-policer policer-name;
mac-validate (loose | strict);
mtu bytes;
no-neighbor-learn;
no-redirects;
policer {
    arp policer-template-name;
    input policer-name;
    output policer-name;
}
primary;
receive-options-packets;
receive-ttl-exceeded;
rpf-check {
    fail-filter filter-name;
    mode loose;
}
sampling {
    (input | output | input output);
}
simple-filter {
    input filter-name;
}
targeted-broadcast {
    forward-and-send-to-re;
    forward-only;
}
unnumbered-address interface-name <destination address>
    <destination-profile profile-name> <preferred-source-address address>;
}

family inet6 {
    address ipv6-address {
        destination destination-address;
        eui-64;
        ndp ipv6-address <l2-interface interface-name> <(mac mac-address |
            multicast-mac multicast-mac-address) <publish>>;
        preferred;
        primary;
        vrrp-inet6-group group-number {
            (accept-data | no-accept-data);
            fast-interval milliseconds;
            inet6-advertise-interval seconds;
            (no-preempt; | ... the following preempt statement ...)
            preempt {
                hold-time seconds;
            }
            priority number;
            track {
                interface interface-name {

```

```
        bandwidth-threshold bits-per-second priority-cost priority;  
        priority-cost priority;  
    }  
    priority-hold-time seconds;  
    route ip-address-prefix/prefix-length routing-instance instance-name  
        priority-cost priority;  
    }  
    virtual-inet6-address [ addresses ];  
    virtual-link-local-address ipv6-address;  
    vrrp-inherit-from {  
        active-group group-number;  
        active-interface interface-name;  
    }  
    }  
    }  
    (dad-disable | no-dad-disable);  
    filter {  
        group filter-group-number;  
        (input filter-name | input-list [ filter-names ] );  
        (output filter-name | output-list [ filter-names ] );  
    }  
    input-hierarchical-policer policer-name;  
    mtu bytes;  
    nd6-stale-time seconds;  
    no-neighbor-learn;  
    policer {  
        input policer-name;  
        output policer-name;  
    }  
    rpf-check {  
        fail-filter filter-name;  
        mode loose;  
    }  
    sampling {  
        (input | output | input output);  
    }  
    unnumbered-address interface-name preferred-source-address address;  
    }  
  
    family iso {  
        address iso-address;  
        mtu bytes;  
    }  
  
    family mlfr-end-to-end {  
        bundle logical-interface-name;  
    }  
  
    family mpls {  
        filter {  
            group filter-group-number;  
            (input filter-name | input-list [ filter-names ] );  
            (output filter-name | output-list [ filter-names ] );  
        }  
    }  
}
```

```

    }
    input-hierarchical-policer policer-name;
    maximum-labels maximum-labels;
    mtu bytes;
    policer {
        input policer-name;
        output policer-name;
    }
}

```

```

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

```

**Related Documentation** • *Notational Conventions Used in Junos OS Configuration Hierarchies*

## schedulers (Interfaces)

<b>Syntax</b>	<code>schedulers <i>number</i>;</code>
<b>Hierarchy Level</b>	[edit interfaces]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.2.
<b>Description</b>	Specify number of schedulers for Ethernet IQ2 PIC port interfaces.
<b>Default</b>	If you omit this statement, the 1024 schedulers are distributed equally over all ports in multiples of 4.
<b>Options</b>	<i>number</i> —Number of schedulers to configure on the port. <b>Range:</b> 1 through 1024
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	• <i>Configuring the Number of Schedulers for Ethernet IQ2 PICs</i>

## CHAPTER 30

# Configuration Statements: Tricolor Marking Policers

- [\[edit class-of-service\] Hierarchy Level on page 572](#)
- [classifiers \(Definition\) on page 576](#)
- [code-points on page 577](#)
- [drop-profile \(Schedulers\) on page 577](#)
- [drop-profile-map \(Schedulers\) on page 578](#)
- [dscp \(Multifield Classifier\) on page 579](#)
- [dscp \(Rewrite Rules\) on page 580](#)
- [dscp-ipv6 \(CoS Rewrite Rules\) on page 581](#)
- [exp on page 582](#)
- [forwarding-class \(BA Classifiers\) on page 583](#)
- [ieee-802.1 \(Rewrite Rules on Logical Interface\) on page 584](#)
- [import \(Classifiers\) on page 585](#)
- [import \(Rewrite Rules\) on page 585](#)
- [inet-precedence \(CoS Rewrite Rules\) on page 586](#)
- [loss-priority \(Scheduler Drop Profiles\) on page 587](#)
- [protocol \(Schedulers\) on page 588](#)
- [rewrite-rules \(Definition\) on page 589](#)
- [schedulers \(CoS\) on page 590](#)
- [tri-color on page 591](#)
- [\[edit firewall\] Hierarchy Level on page 591](#)
- [action on page 603](#)
- [family \(Multifield Classifier\) on page 604](#)
- [filter \(Configuring\) on page 605](#)
- [logical-interface-policer on page 606](#)
- [loss-priority \(Firewall Filter\) on page 607](#)
- [loss-priority \(Simple Firewall Filter\) on page 607](#)

- [policer \(Configuring\)](#) on page 608
- [shared-bandwidth-policer \(Configuring\)](#) on page 609
- [term \(Normal Filter\)](#) on page 610
- [then \(Services CoS\)](#) on page 611
- [three-color-policer \(Applying\)](#) on page 612
- [three-color-policer \(Configuring\)](#) on page 613
- [\[edit interfaces\] Hierarchy Level](#) on page 614
- [filter \(Applying to an Interface\)](#) on page 625
- [input-policer](#) on page 626
- [input-three-color](#) on page 627
- [layer2-policer](#) on page 628
- [output-policer](#) on page 629
- [output-three-color](#) on page 630

---

## [\[edit class-of-service\] Hierarchy Level](#)

---

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

```

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

```

```

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

class-of-service {
    interfaces {
        interface-name {
            excess-bandwidth-share (equal | proportional value);
            input-excess-bandwidth-share (equal | proportional value);
            input-scheduler-map map-name;
        }
    }
}

```

```

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

```

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

**Related Documentation**

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

---

## classifiers (Definition)

---

**Syntax**

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

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

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

**Description** Define a CoS aggregate behavior 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.

**Options** ***classifier-name***—Name of the aggregate behavior classifier.  
***type***—Traffic type: **dscp**, **dscp-ipv6**, **exp**, **ieee-802.1**, **ieee-802.1ad**, **inet-precedence**.

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

**Related Documentation**

- *Overview of BA Classifier Types*
- *Example: Configuring CoS for a PBB Network on MX Series Routers*

## code-points

<b>Syntax</b>	<code>code-points ([ <i>aliases</i> ]   [ <i>bit-patterns</i> ] );</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service classifiers <i>type classifier-name</i> forwarding-class <i>class-name</i> loss-priority <i>level</i>]</code>
<b>Release Information</b>	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 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Specify one or more DSCP code-point aliases or bit sets for association with a forwarding class.
<b>Options</b>	<i>aliases</i> —Name of the DSCP alias.  <i>bit-patterns</i> —Value of the code-point bits, in six-bit binary form.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26</a></li> <li>• <a href="#">Example: Configuring and Applying a Custom DSCP Behavior Aggregate Classifier</a></li> </ul>

## drop-profile (Schedulers)

<b>Syntax</b>	<code>drop-profile <i>profile-name</i>;</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> <a href="#">drop-profile-map</a> <i>loss-priority</i> (any   low   medium-low   medium-high   high) <a href="#">protocol</a> (any   non-tcp   tcp)]</code>
<b>Release Information</b>	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.
<b>Description</b>	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.
<b>Options</b>	<i>profile-name</i> —Name of the drop profile.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199</a></li> <li>• <a href="#">Managing Congestion Using RED Drop Profiles and Packet Loss Priorities on page 200</a></li> </ul>

## drop-profile-map (Schedulers)

---

<b>Syntax</b>	drop-profile-map <b>loss-priority</b> (any   low   medium-low   medium-high   high) <b>protocol</b> (any   non-tcp   tcp) <b>drop-profile (Schedulers)</b> <i>profile-name</i> ;
<b>Hierarchy Level</b>	[edit class-of-service <b>schedulers</b> <i>scheduler-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	Define the loss-priority value for a drop profile.  The statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Default Schedulers Overview on page 145</a></li><li>• <a href="#">Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199</a></li></ul>

## dscp (Multifield Classifier)

<b>Syntax</b>	<code>dscp [0   <i>value</i>];</code>
<b>Hierarchy Level</b>	<code>[edit firewall family <i>family-name</i> filter <i>filter-name</i> term <i>term-name</i> <b>then</b>]</code>
<b>Release Information</b>	Statement introduced in Junos OS Release 7.4.
<b>Description</b>	<p>For M320 and T Series routers, set the DSCP field of incoming or outgoing packets to <b>000000</b>. On the same packets, you can use a behavior aggregate (BA) classifier and a rewrite rule to rewrite the MPLS EXP field.</p> <p>For MX Series routers with MPCs and EX Series switches, the DSCP field can be set from a numeric range.</p> <p>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.</p>
<b>Options</b>	<b>value</b> —For MX Series routers with MPCs, specify the field of incoming or outgoing packets in the range from <b>0</b> through <b>63</b> .
<b>Required Privilege Level</b>	firewall—To view this statement in the configuration. firewall-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Applying Tricolor Marking Policers to Firewall Filters on page 107</a></li> </ul>

## dscp (Rewrite Rules)

---

<b>Syntax</b>	<code>dscp</code> ( <i>rewrite-name</i>   <code>default</code> ) <code>protocol mpls</code> ;
<b>Hierarchy Level</b>	[ <code>edit class-of-service interfaces</code> <i>interface-name</i> <code>unit</code> <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>For IPv4 traffic, apply a Differentiated Services (DiffServ) code point (DSCP) rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"><li>• On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li><li>• On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li></ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [<code>edit class-of-service rewrite-rules dscp</code>] hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p><code>interface</code>—To view this statement in the configuration.</p> <p><code>interface-control</code>—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li><li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li><li>• <a href="#">Rewriting MPLS and IPv4 Packet Headers</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## dscp-ipv6 (CoS Rewrite Rules)

<b>Syntax</b>	<code>dscp-ipv6 (rewrite-name   &lt;default&gt;) protocol mpls;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Support for <b>protocol mpls</b> option introduced in Junos OS Release 10.4R2.
<b>Description</b>	<p>For IPv6 traffic, apply a DSCP rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp-ipv6</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"> <li>On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li> <li>On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li> </ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules dscp-ipv6</b>] hierarchy level.</p> <p><b>default</b>—Default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> <li><a href="#">protocol on page 502</a></li> <li><a href="#">Setting IPv6 DSCP and MPLS EXP Values Independently on page 217</a></li> <li><a href="#">Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 220</a></li> <li><a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li><a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## exp

<b>Syntax</b>	<code>exp (rewrite-name   default) protocol protocol-types;</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS Release 12.2. for ACX series
<b>Description</b>	Apply an MPLS experimental (EXP) rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the <code>[edit class-of-service rewrite-rules exp]</code> hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p>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 <b>mpls-inet-both</b> or <b>mpls-inet-both-non-vpn</b> option at the <code>[edit class-of-service interfaces interface interface-name unit logical-unit-number rewrite-rules exp rewrite-rule-name protocol]</code> hierarchy level. The IP precedence rewrite rules explanation does not apply to ACX Series Universal Access routers.</p> <p>On interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series 3D Universal Edge Routers and EX Series switches, we highly recommend that you configure the <b>default</b> option when you configure a behavior aggregate (BA) classifier that does not include a specific rewrite rule for MPLS packets. Doing so ensures that MPLS exp value is rewritten according to the BA classifier rules configured for forwarding or packet loss priority. This does not apply to ACX Series Universal Access routers.</p> <p><b>protocol-types</b>—Specify one or more protocol matching criteria:</p> <ul style="list-style-type: none"> <li>• <b>mpls-any</b>—Apply to MPLS packets, write MPLS header only.</li> <li>• <b>mpls-inet-both</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header.</li> <li>• <b>mpls-inet-both-non-vpn</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header for only non VPN traffic.</li> </ul>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</a></li> <li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li> </ul>

- [rewrite-rules \(Definition\) on page 504](#)

## forwarding-class (BA Classifiers)

---

<b>Syntax</b>	<code>forwarding-class <i>class-name</i> {     <code>loss-priority level code-points</code> [ <i>aliases</i> ] [ <i>bit-patterns</i> ]; }</code>
<b>Hierarchy Level</b>	[edit class-of-service classifiers <i>type classifier-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Define forwarding class name and option values.
<b>Options</b>	<i>class-name</i> —Name of the forwarding class.  The remaining statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Behavior Aggregate Classifiers on page 36</a></li> </ul>

## ieee-802.1 (Rewrite Rules on Logical Interface)

---

<b>Syntax</b>	ieee-802.1 ( <i>rewrite-name</i>   default) <b>vlan-tag</b> (outer   outer-and-inner);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. <b>vlan-tag</b> statement introduced in Junos OS Release 8.1.
<b>Description</b>	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.
<b>Options</b>	<b>rewrite-name</b> —Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules</b> <b>ieee-802.1</b> ] hierarchy level.  <b>default</b> —The default mapping.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li><li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li><li>• <a href="#">exp on page 377</a></li><li>• <a href="#">exp-push-push-push on page 487</a></li><li>• <a href="#">exp-swap-push-push on page 488</a></li><li>• <a href="#">ieee-802.1ad on page 381</a></li><li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## import (Classifiers)

---

<b>Syntax</b>	<code>import (classifier-name   default);</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service classifiers type classifier-name]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Specify a default or previously defined classifier.
<b>Options</b>	<p><b>classifier-name</b>—Name of the classifier mapping configured at the <code>[edit class-of-service classifiers]</code> hierarchy level.</p> <p><b>default</b>—The default classifier mapping.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Overview of BA Classifier Types</a></li> </ul>

## import (Rewrite Rules)

---

<b>Syntax</b>	<code>import (rewrite-name   default);</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service rewrite-rules type rewrite-name]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Specify a default or previously defined <b>rewrite-rules</b> mapping to import.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a rewrite-rules mapping configured at the <code>[edit class-of-service rewrite-rules]</code> hierarchy level.</p> <p><b>default</b>—The default <b>rewrite-rules</b> mapping.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## inet-precedence (CoS Rewrite Rules)

---

<b>Syntax</b>	<code>inet-precedence (<i>rewrite-name</i>   default);</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <a href="#">rewrite-rules</a> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Apply a IPv4 precedence rewrite rule.
<b>Options</b>	<p><b><i>rewrite-name</i></b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service rewrite-rules <b>inet-precedence</b>] hierarchy level.</p> <p><b>default</b>—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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li><li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## loss-priority (Scheduler Drop Profiles)

<b>Syntax</b>	loss-priority (any   high   low   medium-high   medium-low);
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> drop-profile-map]
<b>Release Information</b>	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.
<b>Description</b>	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.
<b>Options</b>	<b>any</b> —The drop profile applies to packets with any PLP.



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

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

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

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

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

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

<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Default Schedulers Overview on page 145</a></li> <li>• <a href="#">Determining Packet Drop Behavior by Configuring Drop Profile Maps for Schedulers on page 199</a></li> <li>• <a href="#">Configuring Schedulers for Priority Scheduling on page 195</a></li> <li>• <a href="#">Configuring Tricolor Marking on page 92</a></li> <li>• <a href="#">protocol (Schedulers) on page 359</a></li> </ul>
------------------------------	---

## protocol (Schedulers)

---

<b>Syntax</b>	protocol (any   non-tcp   tcp);
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">schedulers</a> <i>scheduler-name</i> <a href="#">drop-profile-map</a> ]
<b>Release Information</b>	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.
<b>Description</b>	Specify the protocol type for the specified scheduler.
<b>Options</b>	<b>any</b> —Accept any protocol type.  <b>non-tcp</b> —(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](#).

---

	<b>tcp</b> —(ACX Series Routers, M Series and T Series (except T4000) routers only) Accept TCP/IP protocol type.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Schedulers on page 146</a></li></ul>

## rewrite-rules (Definition)

<b>Syntax</b>	<pre>rewrite-rules {     type <i>rewrite-name</i>{         import (<i>rewrite-name</i>   default);         forwarding-class <i>class-name</i> {             loss-priority <i>level</i> <i>code-point</i> [ <i>aliases</i> ] [ <i>bit-patterns</i> ];         }     } }</pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. ieee-802.1ad option introduced in Junos OS Release 9.2.
<b>Description</b>	Specify a rewrite-rules mapping for the traffic that passes through all queues on the interface.
<b>Options</b>	<p><i>rewrite-name</i>—Name of a rewrite-rules mapping.</p> <p><i>type</i>—Traffic type.</p> <p><b>Values:</b> dscp, dscp-ipv6, exp, ieee-802.1, ieee-802.1ad, inet-precedence</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <i>Example: Configuring CoS for a PBB Network on MX Series Routers</i></li> <li>• J Series router documentation</li> </ul>

## schedulers (CoS)

---

Syntax	<pre>schedulers {   scheduler-name {     adjust-minimum <i>rate</i>;     adjust-percent <i>percentage</i>;     buffer-size (<i>seconds</i>   percent <i>percentage</i>   remainder   temporal <i>microseconds</i>);     drop-profile-map <i>loss-priority</i> (any   low   medium-low   medium-high   high) <i>protocol</i>       (any   non-tcp   tcp) <i>drop-profile</i> <i>profile-name</i>;     excess-priority [ low   medium-low   medium-high   high   none];     excess-rate (percent <i>percentage</i>   proportion <i>value</i>);     priority <i>priority-level</i>;     shaping-rate (percent <i>percentage</i>   <i>rate</i>);     transmit-rate (percent <i>percentage</i>   <i>rate</i>   remainder) &lt;exact   rate-limit&gt;;   } }</pre>
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	<i>scheduler-name</i> —Name of the scheduler to be configured.  The remaining statements are explained separately.
Required Privilege Level	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"><li>• <a href="#">Schedulers Overview on page 143</a></li><li>• <a href="#">Default Schedulers Overview on page 145</a></li><li>• <a href="#">Configuring Schedulers on page 146</a></li><li>• <a href="#">Configuring a Scheduler</a></li></ul>

## tri-color

<b>Syntax</b>	tri-color;
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced in Junos OS Release 7.4.
<b>Description</b>	For IPv4 packets on M320, MX Series, T Series routers with Enhanced II Flexible PIC Concentrators (FPCs), and EX Series switches, enable two-rate tricolor marking (TCM), as defined in RFC 2698.
<b>Default</b>	If you do not include this statement, tricolor marking is not enabled and the medium packet loss priority (PLP) is not configurable.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Tricolor Marking on page 92</a></li> </ul>

## [edit firewall] Hierarchy Level

Several statements in the **[edit firewall]** hierarchy are valid at numerous locations within the hierarchy. .

- [Common Firewall Actions on page 591](#)
- [Common IP Firewall Actions on page 592](#)
- [Common IPv4 and IPv6 Firewall Actions on page 592](#)
- [Common IP Firewall Match Conditions on page 593](#)
- [Common IPv4 Firewall Match Conditions on page 594](#)
- [Common Layer 2 Firewall Match Conditions on page 594](#)
- [Complete \[edit firewall\] Hierarchy on page 596](#)

## Common Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on page 326 instead of the statements being repeated.

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

The common firewall actions are as follows:

```
count counter-name;
forwarding-class class-name;
```

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

## Common IP Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in [“Complete \[edit firewall\] Hierarchy” on page 326](#) instead of the statements being repeated.

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

The common IP firewall actions are as follows:

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

## Common IPv4 and IPv6 Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in [“Complete \[edit firewall\] Hierarchy” on page 326](#) instead of the statements being repeated.

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

The common IP version 4 (IPv4) and IP version 6 (IPv6) firewall actions are as follows:

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

## Common IP Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in “[Complete \[edit firewall\] Hierarchy](#)” on page 326 instead of the statements being repeated.

- **[edit firewall family inet dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in “[Complete \[edit firewall\] Hierarchy](#)” on page 326)
- **[edit firewall family inet filter *filter-name* term *term-name* from]**
- **[edit firewall family inet6 dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in “[Complete \[edit firewall\] Hierarchy](#)” on page 326)
- **[edit firewall family inet6 filter *filter-name* term *term-name* from]**
- **[edit firewall filter *filter-name* term *term-name* from]**

The common IP firewall match conditions are as follows:

```

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

```

## Common IPv4 Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in [“Complete \[edit firewall\] Hierarchy” on page 326](#) instead of the statements being repeated.

- **[edit firewall family inet dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in [“Complete \[edit firewall\] Hierarchy” on page 326](#))
- **[edit firewall family inet filter *filter-name* term *term-name* from]**
- **[edit firewall filter *filter-name* term *term-name* from]**

The common IPv4 firewall match conditions are as follows:

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

## Common Layer 2 Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in [“Complete \[edit firewall\] Hierarchy” on page 326](#) instead of the statements being repeated.

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

The common Layer 2 firewall match conditions are as follows:

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

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

## Complete [edit firewall] Hierarchy

```

firewall {
  family (any | ethernet-switching | inet | inet6) {
    ... the family subhierarchies appear after the main [edit firewall] hierarchy ...
  }
  filter filter-name {
    accounting-profile [ profile-names ];
    enhanced-mode;
    interface-shared-with;
    interface-specific;
    physical-interface-policer;
    term term-name {
      filter filter-name;
      from {
        ... statements in Common IP Firewall Match Conditions on page 323 AND
        ... statements in Common IPv4 Firewall Match Conditions on page 324 ...
      }
      then {
        ... statements in Common Firewall Actions on page 321 AND
        ... statements in Common IP Firewall Actions on page 322 AND
        ... statements in Common IPv4 and IPv6 Firewall Actions on page 322 ...
      }
    }
  }
}
hierarchical-policer policer-name {
  aggregate {
    if-exceeding {
      bandwidth-limit bps;
      burst-size-limit bytes;
    }
    then {
      discard;
      forwarding-class class-name;
      loss-priority (high | low | medium-high | medium-low);
    }
  }
  logical-interface-policer;
  physical-interface-policer;
  premium {
    if-exceeding {
      bandwidth-limit bps;
      burst-size-limit bytes;
    }
    then {
      discard;
    }
  }
}
shared-bandwidth-policer;
interface-set interface-set-name {
  interface-name;
}
load-balance-group group-name {
  next-hop-group [ group-names ];
}

```

```

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

firewall {
  family any {
    filter filter-name {
      interface-shared;
      term term-name {
        from {
          (forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
          interface interface-name;
          interface-set set-name;
          (loss-priority [ priorities ] | loss-priority-except [ priorities ]);
          (packet-length [ values ] | packet-length-except [ values ]);
        }
        then {
          ... statements in Common Firewall Actions on page 321 PLUS ...
          (accept | discard);
        }
      }
    }
  }
}

```

```
    }  
  }  
}
```

```
firewall {  
  family ethernet-switching {  
    filter filter-name {  
      interface-specific;  
      term term-name {  
        from {  
          destination-address {  
            ip-prefix</prefix-length>;  
          }  
          destination-mac-address {  
            mac-address;  
          }  
          destination-port [ port-names ];  
          destination-prefix-list {  
            list-name;  
          }  
          dot1q-tag [ tag-values ];  
          dot1q-user-priority [ priority-values ];  
          dscp [ code-point-values ];  
          ether-type [ protocol-names ];  
          fragment-flags flag;  
          icmp-code [ codes ];  
          icmp-type [ types ];  
          interface interface-name;  
          is-fragment;  
          precedence [ precedence-names ];  
          protocol [ protocol-names ];  
          source-address {  
            ip-prefix</prefix-length>;  
          }  
          source-mac-address {  
            mac-address;  
          }  
          source-port [ port-names ];  
          source-prefix-list {  
            list-name;  
          }  
          tcp-established;  
          tcp-flags flag;  
          tcp-initial;  
          vlan [ vlan-names ];  
        }  
        then {  
          (accept | discard);  
          analyzer analyzer-name;  
          count counter-name;  
          forwarding-class class-name;  
          interface interface-name;  
          log;  
        }  
      }  
    }  
  }  
}
```

```

        loss-priority (high | low);
        policer policer-name;
        syslog;
        vlan vlan-name;
    }
}
}
}
}

firewall {
    family inet {
        dialer-filter filter-name {
            accounting-profile [ profile-names ];
            term term-name {
                from {
                    ... statements in Common IP Firewall Match Conditions on page 323 AND
                    statements in Common IPv4 Firewall Match Conditions on page 324 EXCEPT
                    FOR ...
                    (ah-spi [ values ] | ah-spi-except [ values ]); # NOT valid at this level
                    (destination-class [ class-names ] |
                     destination-class-except [ class-names ]); # NOT valid at this level
                    interface interface-name; # NOT valid at this level
                    (loss-priority [ priorities ] | loss-priority-except [ priorities ]); # NOT valid at
                     this level
                    service-filter-hit; # NOT valid at this level
                    (source-class [ class-names ] | source-class-except [ class-names ]); # NOT
                     valid at this level
                }
                then {
                    (ignore | note);
                    log;
                    sample;
                    syslog;
                }
            }
        }
    }
    filter filter-name {
        accounting-profile [ profile-names ];
        interface-specific;
        term term-name {
            filter filter-name;
            from {
                ... statements in Common IP Firewall Match Conditions on page 323 AND
                statements in Common IPv4 Firewall Match Conditions on page 324 ...
            }
            then {
                ... statements in Common Firewall Actions on page 321 AND
                statements in Common IP Firewall Actions on page 322 AND
                statements in Common IPv4 and IPv6 Firewall Actions on page 322 ...
            }
        }
    }
    prefix-action name {
        count;
        destination-prefix-length prefix-length;
    }
}

```

```

filter-specific;
policer policer-name;
source-prefix-length prefix-length;
subnet-prefix-length prefix-length;
}
service-filter filter-name {
term term-name {
from {
address {
ip-prefix</prefix-length>;
}
(ah-spi [ values ] | ah-spi-except [ values ]);
destination-address {
ip-prefix</prefix-length>;
}
(destination-port [ port-names ] | destination-port-except [ port-names ]);
destination-prefix-list {
list-name;
}
(esp-spi [ values ] | esp-spi-except [ values ]);
first-fragment;
fragment-flags flag;
(fragment-offset [ offsets ] | fragment-offset-except [ offsets ]);
(interface-group [ group-names ] | interface-group-except [ group-names ]);
(ip-options [ option-names ] | ip-options-except [ option-names ]);
is-fragment;
(loss-priority [ priorities ] | loss-priority-except [ priorities ]);
(port [ port-names ] | port-except [ port-names ]);
prefix-list {
list-name;
}
(protocol [ protocol-names ] | protocol-except [ protocol-names ]);
source-address {
ip-prefix</prefix-length>;
}
(source-port [ port-names ] | source-port-except [ port-names ]);
source-prefix-list {
list-name;
}
tcp-flags flag-name;
}
then {
count counter-name;
log;
port-mirror;
sample;
(service | skip);
}
}
}
simple-filter filter-name {
term term-name {
from {
destination-address ip-prefix</prefix-length>;
destination-port port-name;
forwarding-class [ class-names ];

```

```

        protocol protocol-name;
        source-address ip-prefix < / prefix-length >;
        source-port port-name;
    }
    then {
        forwarding-class class-name;
        loss-priority (high | low | medium-high | medium-low);
        policer policer-name;
    }
}
}
}
}

firewall {
    family inet6 {
        dialer-filter filter-name {
            accounting-profile [ profile-names ];
            term term-name {
                from {
                    ... statements in Common IP Firewall Match Conditions on page 323 PLUS ...
                    (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
                    ... BUT NOT ...
                    (destination-class [ class-names ] |
                     destination-class-except [ class-names ]); # NOT valid at this level
                    (forwarding-class [ class-names ] |
                     forwarding-class-except [ class-names ]); # NOT valid at this level
                    interface interface-name; # NOT valid at this level
                    (interface-group [ group-names ] | interface-group-except [ group-names ]); #
                     NOT valid at this level
                    (loss-priority [ priorities ] | loss-priority-except [ priorities ]); # NOT valid at
                     this level
                    service-filter-hit; # NOT valid at this level
                    (source-class [ class-names ] | source-class-except [ class-names ]); # NOT
                     valid at this level
                    tcp-established; # NOT valid at this level
                    tcp-flags flag; # NOT valid at this level
                    tcp-initial; # NOT valid at this level
                }
                then {
                    (ignore | note);
                    log;
                    sample;
                    syslog;
                }
            }
        }
    }
    filter filter-name {
        accounting-profile [ profile-names ];
        interface-specific;
        term term-name {
            filter filter-name;
            from {
                ... statements in Common IP Firewall Match Conditions on page 323 PLUS ...
                (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
                (traffic-class [ code-point-values ] | traffic-class-except [ code-point-values ]);
            }
        }
    }
}

```

```

    }
    then {
        ... statements in Common Firewall Actions on page 321 AND
        statements in Common IP Firewall Actions on page 322 PLUS ...
        (accept | discard | reject <address-unreachable | administratively-prohibited |
        beyond-scope | fragmentation-needed | no-route | port-unreachable |
        tcp-reset>);
    }
}
}
service-filter filter-name {
    term term-name {
        from {
            address {
                ip-prefix </prefix-length>;
            }
            (ah-spi [ values ] | ah-spi-except [ values ]);
            destination-address {
                ip-prefix </prefix-length>;
            }
            (destination-port [ port-names ] | destination-port-except [ port-names ]);
            destination-prefix-list {
                list-name;
            }
            (esp-spi [ values ] | esp-spi-except [ values ]);
            (interface-group [ group-names ] | interface-group-except [ group-names ]);
            (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
            (port [ port-names ] | port-except [ port-names ]);
            prefix-list {
                list-name;
            }
            source-address {
                ip-prefix </prefix-length>;
            }
            (source-port [ port-names ] | source-port-except [ port-names ]);
            source-prefix-list {
                list-name;
            }
            tcp-flags flag-name;
        }
        then {
            count counter-name;
            log;
            port-mirror;
            sample;
            (service | skip);
        }
    }
}
}
}

```

- Related Documentation**
- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## action

<b>Syntax</b>	<pre>action {   loss-priority high then discard; }</pre>
<b>Hierarchy Level</b>	<p>[edit dynamic-profiles <i>profile-name</i> firewall <b>three-color-policer</b> <i>name</i>],          [edit firewall <b>three-color-policer</b> <i>name</i>],          [edit logical-systems <i>logical-system-name</i> firewall <b>three-color-policer</b> <i>name</i>]</p>
<b>Release Information</b>	<p>Statement introduced in Junos OS Release 8.2.          Logical systems support introduced in Junos OS Release 9.3.          Support at the [edit dynamic-profiles ... <b>three-color-policer</b>] hierarchy level introduced in Junos OS Release 11.4.          Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p>
<b>Description</b>	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.

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

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

## family (Multifield Classifier)

---

**Syntax**    `family family-name {  
                  filter filter-name {  
                    term term-name {  
                      ... 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.

**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 70](#)

## filter (Configuring)

<b>Syntax</b>	<pre>filter <i>filter-name</i> {     accounting-profile <i>name</i>;     enhanced-mode;     fast-lookup-filter;     filter-list-template;     interface-shared;     interface-specific;     physical-interface-filter;     promote gre-key;     term <i>term-name</i> {         ... term configuration ...     } }</pre>
<b>Hierarchy Level</b>	<p>[edit firewall family <i>family-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> firewall family <i>family-name</i>]</p>
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Logical systems support introduced in Junos OS Release 9.3.</p> <p><b>physical-interface-filter</b> statement introduced in Junos OS Release 9.6.</p> <p>Support for the <b>interface-shared</b> statement introduced in Junos OS Release 12.2.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p>
<b>Description</b>	<p>Configure firewall filters.</p>
<b>Options</b>	<p><b><i>filter-name</i></b>—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 <b>__.*__</b> (beginning and ending with underscores) or <b>__.*</b> (beginning with an underscore).</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>firewall—To view this statement in the configuration.</p> <p>firewall-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Guidelines for Configuring Firewall Filters</i></li> <li>• <i>Guidelines for Applying Firewall Filters</i></li> <li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li> <li>• <i>Using Multifield Classifiers to Set Packet Loss Priority</i></li> <li>• <i>simple-filter (Configuring)</i></li> </ul>

## logical-interface-policer

<b>Syntax</b>	logical-interface-policer;
<b>Hierarchy Level</b>	<p>[edit dynamic-profiles <i>profile-name</i> firewall <b>policer</b> <i>policer-name</i>],</p> <p>[edit dynamic-profiles <i>profile-name</i> firewall <b>three-color-policer</b> <i>name</i>],</p> <p>[edit firewall atm-policer <i>atm-policer-name</i>]</p> <p>[edit firewall <b>policer</b> <i>policer-name</i>],</p> <p>[edit firewall policer <i>policer-template-name</i>],</p> <p>[edit firewall <b>three-color-policer</b> <i>policer-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> firewall <b>policer</b> <i>policer-name</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> firewall <b>three-color-policer</b> <i>name</i>]</p>
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Support at the [edit firewall <b>three-color-policer</b> <i>policer-name</i>] hierarchy level introduced in Junos OS Release 8.2.</p> <p>Logical systems support introduced in Junos OS Release 9.3.</p> <p>Support at the [edit dynamic-profiles ... <b>policer</b> <i>policer-name</i>] and [edit dynamic-profiles ... <b>three-color-policer</b> <i>name</i>] hierarchy levels introduced in Junos OS Release 11.4.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p>
<b>Description</b>	Configure a logical interface policer.



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

<b>Required Privilege Level</b>	<p>firewall—To view this statement in the configuration.</p> <p>firewall-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Two-Color and Three-Color Logical Interface Policers</i></li> <li>• <i>Traffic Policer Types</i></li> <li>• <a href="#">Configuring and Applying Tricolor Marking Policers on page 102</a></li> <li>• <a href="#">action on page 603</a></li> <li>• <i>Configuring Gigabit Ethernet Two-Color and Tricolor Policers</i></li> <li>• <i>action</i></li> </ul>

## loss-priority (Firewall Filter)

---

<b>Syntax</b>	loss-priority (high   low);
<b>Hierarchy Level</b>	[edit firewall family <i>family-name</i> filter <i>filter-name</i> term <i>term-name</i> <b>then</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Set the loss priority of incoming packets.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li></ul>

## loss-priority (Simple Firewall Filter)

---

<b>Syntax</b>	loss-priority (high   low   medium);
<b>Hierarchy Level</b>	[edit firewall family <i>family-name</i> simple-filter <i>filter-name</i> term <i>term-name</i> <b>then</b> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 7.6.
<b>Description</b>	Set the loss priority of incoming packets.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li></ul>

## policer (Configuring)

<b>Syntax</b>	<pre> policer <i>policer-name</i> {     filter-specific;     if-exceeding {         bandwidth-limit <i>bps</i>;         bandwidth-percent <i>number</i>;         burst-size-limit <i>bytes</i>;     }     logical-bandwidth-policer;     logical-interface-policer;     physical-interface-policer;     shared-bandwidth-policer;     then {         <i>policer-action</i>;     } } </pre>
<b>Hierarchy Level</b>	<p>[edit dynamic-profiles <i>profile-name</i> firewall],  [edit firewall],  [edit logical-systems <i>logical-system-name</i> firewall]</p>
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>The <b>out-of-profile</b> policer action added in Junos OS Release 8.1.</p> <p>The <b>logical-bandwidth-policer</b> statement added in Junos OS Release 8.2.</p> <p>Logical systems support introduced in Junos OS Release 9.3.</p> <p>The <b>physical-interface-policer</b> statement introduced in Junos OS Release 9.6.</p> <p>The <b>shared-bandwidth-policer</b> statement added in Junos OS Release 11.2.</p> <p>Support at the [edit dynamic-profiles ... firewall] hierarchy level introduced in Junos OS Release 11.4.</p> <p>Statement introduced in Junos OS Release 12.3R2 for EX Series switches.</p>
<b>Description</b>	<p>Configure policer rate limits and actions. When included at the [edit firewall] hierarchy level, the <b>policer</b> 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 <b>policer-action</b> modifier in the <b>then</b> statement in a firewall filter term or on an interface.</p>
<b>Options</b>	<p><b><i>policer-action</i></b>—One or more actions to take:</p> <ul style="list-style-type: none"> <li>• <b>discard</b>—Discard traffic that exceeds the rate limits.</li> <li>• <b>forwarding-class <i>class-name</i></b>—Specify the particular forwarding class.</li> <li>• <b>loss-priority</b>—Set the packet loss priority (PLP) to <b>low</b>, <b>medium-low</b>, <b>medium-high</b>, or <b>high</b>.</li> </ul> <p><b><i>policer-name</i></b>—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 <b>_.*</b>.</p> <p><b>then</b>—Actions to take on matching packets.</p>

The remaining statements are explained separately.

<b>Required Privilege Level</b>	firewall—To view this statement in the configuration. firewall-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Bandwidth Policer Overview</i></li> <li>• <i>Configuring Firewall Filters and Policers for VPLS</i></li> <li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li> <li>• <i>Logical Interface (Aggregate) Policer Overview</i></li> <li>• <i>Physical Interface Policer Overview</i></li> <li>• <i>Statement Hierarchy for Configuring Policers</i></li> <li>• <i>Single-Rate Two-Color Policer Overview</i></li> <li>• <i>Using Multifield Classifiers to Set Packet Loss Priority</i></li> <li>• <a href="#">filter (Configuring) on page 451</a></li> <li>• <a href="#">priority (Schedulers) on page 358</a></li> </ul>

## shared-bandwidth-policer (Configuring)

<b>Syntax</b>	shared-bandwidth-policer;
<b>Hierarchy Level</b>	[edit firewall <a href="#">policer</a> <i>policer-name</i> ], [edit firewall <a href="#">three-color-policer</a> <i>policer-name</i> ], [edit firewall hierarchical-policer <i>policer-name</i> ]
<b>Release Information</b>	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.
<b>Description</b>	<p>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.</p> <p>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.</p>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Policer Support for Aggregated Ethernet Bundle Overview on page 269</a></li> </ul>

## term (Normal Filter)

---

<b>Syntax</b>	<pre>term <i>term-name</i> {     from {         <i>match-conditions</i>;     }     then {         forwarding-class <i>class-name</i>;         loss-priority (high   low);         three-color-policer {             (single-rate   two-rate) <i>policer-name</i>;         }     } }</pre>
<b>Hierarchy Level</b>	[edit firewall family <i>family-name</i> <b>filter</b> <i>filter-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Define a firewall filter term.
<b>Options</b>	<p><b>from</b>—Match packet fields to values. If not included, all packets are considered to match and the actions and action modifiers in the <b>then</b> statement are taken.</p> <p><b>match-conditions</b>—One or more conditions to use to make a match.</p> <p><b>term-name</b>—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 (" ").</p> <p><b>then</b>—Actions to take on matching packets. If not included and a packet matches all the conditions in the <b>from</b> statement, the packet is accepted. For CoS, only the actions listed are allowed. These statements are explained separately.</p>
<b>Required Privilege Level</b>	firewall—To view this statement in the configuration. firewall-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Multifield Classifiers on page 70</a></li></ul>

## then (Services CoS)

<b>Syntax</b>	<pre> then {   application-profile <i>profile-name</i>;   dscp (<i>alias</i>   <i>bits</i>);   forwarding-class <i>class-name</i>;   syslog;   (reflexive   reverse) {     application-profile <i>profile-name</i>;     dscp (<i>alias</i>   <i>bits</i>);     forwarding-class <i>class-name</i>;     syslog;   } } </pre>
<b>Hierarchy Level</b>	[edit services cos rule <i>rule-name</i> term <i>term-name</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.1.
<b>Description</b>	<p>Define the CoS term actions.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Configuring Actions in a CoS Rule</i></li> <li>• <i>Configuring Actions in CoS Rules</i></li> </ul>

## three-color-policer (Applying)

---

<b>Syntax</b>	<pre>three-color-policer {     (single-rate   two-rate) <i>policer-name</i>; }</pre>
<b>Hierarchy Level</b>	[edit firewall family <i>family-name</i> filter <i>filter-name</i> term <i>term-name</i> then] [edit logical-systems <i>logical-system-name</i> firewall family <i>family-name</i> <b>filter</b> <i>filter-name</i> term <i>term-name</i> then]
<b>Release Information</b>	Statement introduced in Junos OS Release 7.4. <b>single-rate</b> 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.
<b>Description</b>	For M320 and T Series routers with Enhanced II Flexible PIC Concentrators (FPCs) and the T640 router with Enhanced Scaling FPC4, apply a tricolor marking policer.
<b>Options</b>	<b>single-rate</b> —Named tricolor policer is a single-rate policer.  <b>two-rate</b> —Named tricolor policer is a two-rate policer.  <b><i>policer-name</i></b> —Name of a tricolor policer.
<b>Required Privilege Level</b>	firewall—To view this statement in the configuration. firewall-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Applying Tricolor Marking Policers to Firewall Filters on page 107</a></li><li>• <i>Firewall Filter Nonterminating Actions</i></li><li>• <i>Three-Color Policer Configuration Overview</i></li></ul>

## three-color-policer (Configuring)

<b>Syntax</b>	<pre> three-color-policer <i>policer-name</i>   <i>uid</i> {   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 <i>bytes</i>;     committed-information-rate <i>bps</i>;     excess-burst-size <i>bytes</i>;   }   two-rate {     (color-aware   color-blind);     committed-burst-size <i>bytes</i>;     committed-information-rate <i>bps</i>;     peak-burst-size <i>bytes</i>;     peak-information-rate <i>bps</i>;   } } </pre>
<b>Hierarchy Level</b>	[edit dynamic-profiles <i>profile-name</i> firewall], [edit firewall], [edit logical-systems <i>logical-system-name</i> firewall]
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>The <b>action</b> and <b>single-rate</b> statements added in Junos OS Release 8.2.</p> <p>Logical systems support introduced in Junos OS Release 9.3.</p> <p>Support at the [edit dynamic-profiles ... firewall] hierarchy level introduced in Junos OS Release 11.4.</p>
<b>Description</b>	Configure a three-color policer.
<b>Options</b>	<p><b><i>policer-name</i></b>—Name of the three-color policer. Reference this name when you apply the policer to an interface.</p> <p><b><i>uid</i></b>—When you configure a policer at the [edit dynamic-profiles] hierarchy level, you must assign a variable UID as the policer name.</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	firewall—To view this statement in the configuration. firewall-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Statement Hierarchy for Configuring Policers</i></li> <li>• <a href="#">Configuring and Applying Tricolor Marking Policers on page 102</a></li> <li>• <i>Three-Color Policer Configuration Guidelines</i></li> </ul>

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

---

## [edit interfaces] Hierarchy Level

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

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

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

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

```

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

```

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

```

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

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

```

```

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

```

```

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

```

```

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

```

```

interface-name {
    unit logical-unit-number {
        disable;
        accept-source-mac {
            mac-address mac-address {
                policer {
                    input policer-name;
                    output policer-name;
                }
            }
        }
    }
}
accounting-profile name;

```

```

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

unit logical-unit-number {

```

```

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

```

```

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

```

```
        unnumbered-address interface-name preferred-source-address address;  
    }  
  
    family iso {  
        address iso-address;  
        mtu bytes;  
    }  
  
    family mlfr-end-to-end {  
        bundle logical-interface-name;  
    }  
  
    family mpls {  
        filter {  
            group filter-group-number;  
            (input filter-name | input-list [ filter-names ] );  
            (output filter-name | output-list [ filter-names ] );  
        }  
        input-hierarchical-policer policer-name;  
        maximum-labels maximum-labels;  
        mtu bytes;  
        policer {  
            input policer-name;  
            output policer-name;  
        }  
    }  
  
    family vpls {  
        core-facing;  
        filter {  
            group filter-group-number;  
            (input filter-name | input-list [ filter-names ] );  
            (output filter-name | output-list [ filter-names ] );  
        }  
        policer {  
            input policer-name;  
            output policer-name;  
        }  
    }  
}  
}
```

**Related  
Documentation**

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## filter (Applying to an Interface)

<b>Syntax</b>	<pre>filter {     input <i>filter-name</i>;     output <i>filter-name</i>; }</pre>
<b>Hierarchy Level</b>	[edit interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i> <b>family</b> <i>family</i> ], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> <b>unit</b> <i>logical-unit-number</i> <b>family</b> <i>family</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Apply a filter to an interface. You can also use filters for encrypted traffic. When you configure filters, you can configure the family <b>inet</b> , <b>inet6</b> , <b>mpls</b> , or <b>vppls</b> only.
<b>Options</b>	<p><b>input <i>filter-name</i></b>—Name of one filter to evaluate when packets are received on the interface.</p> <p><b>output <i>filter-name</i></b>—Name of one filter to evaluate when packets are transmitted on the interface.</p>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">simple-filter on page 468</a></li> <li>• <a href="#">Applying Firewall Filter Tricolor Marking Policers to Interfaces</a></li> <li>• <a href="#">Example: Classifying Packets Based on Their Destination Address</a></li> <li>• <a href="#">Example: Configuring and Verifying a Complex Multifield Filter on page 74</a></li> <li>• <a href="#">Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets</a></li> <li>• <a href="#">Example: Configuring a Simple Filter on page 254</a></li> <li>• <a href="#">Example: Configuring a Logical Bandwidth Policer on page 89</a></li> <li>• <a href="#">Configuring Two-Color Policers and Shaping Rate Changes on page 86</a></li> </ul>

## input-policer

---

<b>Syntax</b>	<code>input-policer <i>policer-name</i>;</code>
<b>Hierarchy Level</b>	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> layer2-policer]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.2. Statement introduced in Junos OS Release 12.3R2 for EX Series switches.
<b>Description</b>	Associate a Layer 2 policer with a logical interface. The <b>input-policer</b> and <b>input-three-color</b> statements are mutually exclusive.
<b>Options</b>	<i>policer-name</i> —Name of the policer that you define at the [edit firewall] hierarchy level.
<b>Usage Guidelines</b>	See “ <a href="#">Applying Layer 2 Policers to Gigabit Ethernet Interfaces</a> ” on page 270.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Two-Color and Three-Color Policers at Layer 2</i></li><li>• <a href="#">Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 270</a></li><li>• <i>Configuring a Gigabit Ethernet Policer</i></li><li>• <a href="#">input-three-color on page 627</a></li><li>• <a href="#">layer2-policer on page 628</a></li><li>• <a href="#">logical-interface-policer on page 606</a></li><li>• <a href="#">output-policer on page 629</a></li><li>• <a href="#">output-three-color on page 630</a></li></ul>

## input-three-color

---

<b>Syntax</b>	<code>input-three-color <i>policer-name</i>;</code>
<b>Hierarchy Level</b>	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> layer2-policer]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.2. Statement introduced in Junos OS Release 12.3R2 for EX Series switches.
<b>Description</b>	Apply a single-rate or two-rate three-color policer to the Layer 2 input traffic at the logical interface. The <b>input-three-color</b> and <b>input-policer</b> statements are mutually exclusive.
<b>Options</b>	<i>policer-name</i> —Name of the single-rate or two-rate three-color policer.
<b>Usage Guidelines</b>	See <a href="#">“Applying Layer 2 Policers to Gigabit Ethernet Interfaces” on page 270</a> .
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Two-Color and Three-Color Policers at Layer 2</a></li> <li>• <a href="#">Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 270</a></li> <li>• <a href="#">Configuring a Gigabit Ethernet Policer</a></li> <li>• <a href="#">input-policer on page 626</a></li> <li>• <a href="#">layer2-policer on page 628</a></li> <li>• <a href="#">logical-interface-policer on page 606</a></li> <li>• <a href="#">output-policer on page 629</a></li> <li>• <a href="#">output-three-color on page 630</a></li> </ul>

## layer2-policer

---

<b>Syntax</b>	<pre>layer2-policer {     input-policer <i>policer-name</i>;     input-three-color <i>policer-name</i>;     output-policer <i>policer-name</i>;     output-three-color <i>policer-name</i>; }</pre>
<b>Hierarchy Level</b>	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> ], [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.2. Statement introduced in Junos OS Release 12.3R2 for EX Series switches.
<b>Description</b>	<p>For 1-Gigabit Ethernet and 10-Gigabit Ethernet IQ2 and IQ2-E interfaces on M Series and T Series routers, apply Layer 2 logical interface policers. The following policers are supported:</p> <ul style="list-style-type: none"><li>• Two-color</li><li>• Single-rate tricolor marking (srTCM)</li><li>• Two-rate tricolor marking (trTCM)</li></ul> <p>Two-color and tricolor policers are configured at the <b>[edit firewall]</b> hierarchy level.</p>
<b>Options</b>	<p><b>input-policer <i>policer-name</i></b>—Two-color input policer to associate with the interface. This statement is mutually exclusive with the <b>input-three-color</b> statement.</p> <p><b>input-three-color <i>policer-name</i></b>—Tricolor input policer to associate with the interface. This statement is mutually exclusive with the <b>input-policer</b> statement.</p> <p><b>output-policer <i>policer-name</i></b>—Two-color output policer to associate with the interface. This statement is mutually exclusive with the <b>output-three-color</b> statement.</p> <p><b>output-three-color <i>policer-name</i></b>—Tricolor output policer to associate with the interface. This statement is mutually exclusive with the <b>output-policer</b> statement.</p>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 270</a></li><li>• <i>Configuring Gigabit Ethernet Two-Color and Tricolor Policers</i></li><li>• <i>Class of Service Feature Guide for Routing Devices</i></li><li>• <i>Routing Policies, Firewall Filters, and Traffic Policers Feature Guide for Routing Devices</i></li></ul>

## output-policer

---

<b>Syntax</b>	<code>output-policer <i>policer-name</i>;</code>
<b>Hierarchy Level</b>	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> layer2-policer]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.2. Statement introduced in Junos OS Release 12.3R2 for EX Series switches.
<b>Description</b>	Apply a single-rate two-color policer to the Layer 2 output traffic at the logical interface. The <b>output-policer</b> and <b>output-three-color</b> statements are mutually exclusive.
<b>Options</b>	<b><i>policer-name</i></b> —Name of the single-rate two-color policer that you define at the [edit firewall] hierarchy level.
<b>Usage Guidelines</b>	See “ <a href="#">Applying Layer 2 Policers to Gigabit Ethernet Interfaces</a> ” on page 270.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Two-Color and Three-Color Policers at Layer 2</i></li> <li>• <a href="#">Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 270</a></li> <li>• <i>Configuring a Gigabit Ethernet Policer</i></li> <li>• <a href="#">input-policer on page 626</a></li> <li>• <a href="#">input-three-color on page 627</a></li> <li>• <a href="#">layer2-policer on page 628</a></li> <li>• <a href="#">logical-interface-policer on page 606</a></li> <li>• <a href="#">output-three-color on page 630</a></li> </ul>

## output-three-color

---

<b>Syntax</b>	<code>output-three-color <i>policer-name</i>;</code>
<b>Hierarchy Level</b>	[edit interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <a href="#">layer2-policer</a> ] [edit logical-systems <i>logical-system-name</i> interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <a href="#">layer2-policer</a> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.2. Statement introduced in Junos OS Release 12.3R2 for EX Series switches.
<b>Description</b>	Apply a single-rate or two-rate three-color policer to the Layer 2 output traffic at the logical interface. The <b>output-three-color</b> and <b>output-policer</b> statements are mutually exclusive.
<b>Options</b>	<b><i>policer-name</i></b> —Name of the single-rate or two-rate three-color policer.
<b>Usage Guidelines</b>	See “ <a href="#">Applying Layer 2 Policers to Gigabit Ethernet Interfaces</a> ” on page 270.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Two-Color and Three-Color Policers at Layer 2</i></li><li>• <a href="#">Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 270</a></li><li>• <i>Configuring a Gigabit Ethernet Policer</i></li><li>• <a href="#">input-three-color on page 627</a></li><li>• <a href="#">input-policer on page 626</a></li><li>• <a href="#">layer2-policer on page 628</a></li><li>• <a href="#">logical-interface-policer on page 606</a></li><li>• <a href="#">output-policer on page 629</a></li></ul>

# Configuration Statements: Tunnels CoS

- [\[edit class-of-service\] Hierarchy Level](#) on page 631
- [code-point](#) on page 635
- [dscp \(Rewrite Rules\)](#) on page 636
- [dscp-ipv6 \(CoS Rewrite Rules\)](#) on page 637
- [exp](#) on page 638
- [exp-push-push-push](#) on page 639
- [exp-swap-push-push](#) on page 640
- [forwarding-class \(BA Classifiers\)](#) on page 641
- [ieee-802.1 \(Rewrite Rules on Logical Interface\)](#) on page 642
- [import \(Rewrite Rules\)](#) on page 643
- [inet-precedence \(CoS Rewrite Rules\)](#) on page 643
- [interfaces](#) on page 644
- [loss-priority \(BA Classifiers\)](#) on page 646
- [protocol \(Rewrite Rules\)](#) on page 647
- [rewrite-rules \(Definition\)](#) on page 648
- [rewrite-rules \(Interfaces\)](#) on page 649
- [unit](#) on page 651
- [\[edit interfaces\] Hierarchy Level](#) on page 651
- [copy-tos-to-outer-ip-header](#) on page 663

## [\[edit class-of-service\] Hierarchy Level](#)

---

```
class-of-service {
  classifiers {
    type classifier-name {
      forwarding-class class-name {
        loss-priority (high | low | medium-high | medium-low) code-points [ aliases bits ];
      }
      import (classifier-name | default);
    }
  }
  code-point-aliases {
```

```

(dscp | dscp-ipv6 | exp | ieee-802.1 | ieee-802.1ad | inet-precedence) {
    alias-name bits;
}
}
drop-profiles {
    profile-name {
        fill-level percentage drop-probability percentage;
        interpolate {
            drop-probability value;
            fill-level value;
        }
    }
}
fabric {
    scheduler-map {
        priority (high | low) scheduler scheduler-name;
    }
}
forwarding-class-map {
    map-name {
        class class-name queue-num queue-number <restricted-queue queue-number>;
    }
}
forwarding-classes {
    class class-name policing-priority (normal | premium) queue-num queue-number
        priority (high | low);
    queue queue-number class-name policing-priority (normal | premium) priority (high |
        low);
}
forwarding-policy {
    class class-name {
        classification-override {
            forwarding-class class-name;
        }
    }
    next-hop-map map-name {
        forwarding-class class-name {
            discard;
            lsp-next-hop [ lsp-regular-expressions ];
            next-hop [ next-hop-names ];
            non-lsp-next-hop;
        }
    }
}
fragmentation-maps {
    map-name {
        forwarding-class class-name {
            drop-timeout milliseconds;
            fragment-threshold bytes;
            multilink-class number;
            no-fragmentation;
        }
    }
}
host-outbound-traffic {
    dscp-code-point value;
}

```

```

forwarding-class class-name;
ieee-802.1 {
    default value;
    rewrite-rules;
}
tcp {
    raise-internet-control-priority;
}
}
interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
}
}
restricted-queues {
    forwarding-class class-name queue-number;
}
rewrite-rules {
    (dscp | dscp-ipv6 | exp | frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
    rewrite-rule {
        forwarding-class class-name {
            loss-priority level code-point (alias | bits);
        }
        import (rewrite-rule | default);
    }
}
}
routing-instances routing-instance-name {
    classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
    }
}
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}
schedulers {
    scheduler-name {
        adjust-minimum value;
        adjust-percent value;
        buffer-size (exact | percent percentage | remainder);
        drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
            protocol any;
        excess-priority (high | low | medium-high | medium-low);
        excess-rate (percent percentage | proportion proportion);
        priority (high | low | medium-high | medium-low | strict-high);
        shaping-rate (bps | percent percentage | burst-size size);
        transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
    }
}
}
traceoptions {
    file <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
}

```

```
    flag flag;  
    no-remote-trace;  
}  
traffic-control-profiles {  
    profile-name {  
        adjust-minimum rate;  
        delay-buffer-rate (bps | cps cps | percent percentage);  
        excess-rate (percent percentage | proportion value);  
        guaranteed-rate (bps | percent percentage) <burst-size bytes>;  
        overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;  
        scheduler-map map-name;  
        shaping-rate (bps | percent percentage) <burst-size bytes>;  
    }  
}  
tri-color;  
}  
  
class-of-service {  
    interfaces {  
        interface-name {  
            excess-bandwidth-share (equal | proportional value);  
            input-excess-bandwidth-share (equal | proportional value);  
            input-scheduler-map map-name;  
            input-shaping-rate bps;  
            input-traffic-control-profile profile-name;  
            output-forwarding-class-map map-name;  
            output-traffic-control-profile profile-name;  
            scheduler-map map-name;  
            scheduler-map-chassis (map-name | derived);  
            shaping-rate bps;  
            unit (logical-unit-number | *) {  
                classifiers {  
                    dscp (classifier-name | default) {  
                        family [ inet mpls ];  
                    }  
                    dscp-ipv6 (classifier-name | default) {  
                        family [ inet mpls ];  
                    }  
                    exp (classifier-name | default);  
                    ieee-208.1 (classifier-name | default) <vlan-tag (inner | outer)>;  
                    ieee-208.1ad (classifier-name | default);  
                    inet-precedence (classifier-name | default);  
                }  
                forwarding-class class-name;  
                input-scheduler-map map-name;  
                input-shaping-rate bps;  
                input-traffic-control-profile profile-name shared-instance instance-name;  
                loss-priority-maps {  
                    (map-name | default);  
                }  
                loss-priority-rewrites {  
                    (map-name | default);  
                }  
                output-forwarding-class-map map-name;  
                output-traffic-control-profile profile-name shared-instance instance-name;  
                rewrite-rules {  
                    <...>  
                }  
            }  
        }  
    }  
}
```

```

dscp (rule-name | default) <protocol mpls>;
dscp-ipv6 (rule-name | default);
exp (rule-name | default) <protocol [ mpls-any | mpls-inet-both |
    mpls-inet-both-non-vpn ]>;
exp-push-push-push default;
exp-swap-push-push default;
ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
inet-precedence (rewrite-name | default) <protocol mpls>;
}
scheduler-map map-name;
shaping-rate bps;
translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
    to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
}
}
interface-set interface-set-name {
    excess-bandwidth-share (equal | proportional value);
    input-excess-bandwidth-share (equal | proportional value);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
}
}
}

```

**Related  
Documentation**

- [Notational Conventions Used in Junos OS Configuration Hierarchies](#)

## code-point

<b>Syntax</b>	<code>code-point [ <i>aliases</i> ] [ <i>bit-patterns</i> ];</code>
<b>Hierarchy Level</b>	[edit class-of-service rewrite-rules <i>type</i> <i>rewrite-name</i> <b>forwarding-class</b> <i>class-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Specify one or more code-point aliases or bit sets for association with a forwarding class.
<b>Options</b>	<p><i>aliases</i>—Name of each alias.</p> <p><i>bit-patterns</i>—Value of the code-point bits, in decimal form.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## dscp (Rewrite Rules)

---

<b>Syntax</b>	<code>dscp (rewrite-name   default) protocol mpls;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>For IPv4 traffic, apply a Differentiated Services (DiffServ) code point (DSCP) rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"><li>• On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li><li>• On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li></ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules dscp</b>] hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li><li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li><li>• <a href="#">Rewriting MPLS and IPv4 Packet Headers</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## dscp-ipv6 (CoS Rewrite Rules)

<b>Syntax</b>	<code>dscp-ipv6 (rewrite-name   &lt;default&gt;) protocol mpls;</code>
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Support for <b>protocol mpls</b> option introduced in Junos OS Release 10.4R2.
<b>Description</b>	<p>For IPv6 traffic, apply a DSCP rewrite rule.</p> <p>Logical interfaces do not support multiple <b>dscp-ipv6</b> rewrite rules for the same protocol.</p> <p>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:</p> <ul style="list-style-type: none"> <li>On M320 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs when used with the Enhanced III FPC.</li> <li>On M120 routers, DSCP rewrite is supported on IQ, IQ2, IQE, and IQ2E PICs.</li> </ul> <p>DSCP and DCSP IPv6 rewrite rules are supported on MIC and MPC interfaces on MX Series routers.</p> <p>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.</p>
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules dscp-ipv6</b>] hierarchy level.</p> <p><b>default</b>—Default mapping.</p> <p><b>protocol mpls</b>—(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.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">protocol on page 502</a></li> <li>• <a href="#">Setting IPv6 DSCP and MPLS EXP Values Independently on page 217</a></li> <li>• <a href="#">Configuring DSCP Values for IPv6 Packets Entering the MPLS Tunnel on page 220</a></li> <li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## exp

<b>Syntax</b>	<code>exp (rewrite-name   default) protocol protocol-types;</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced before Junos OS Release 12.2. for ACX series
<b>Description</b>	Apply an MPLS experimental (EXP) rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the <code>[edit class-of-service rewrite-rules exp]</code> hierarchy level.</p> <p><b>default</b>—The default mapping.</p> <p>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 <b>mpls-inet-both</b> or <b>mpls-inet-both-non-vpn</b> option at the <code>[edit class-of-service interfaces interface interface-name unit logical-unit-number rewrite-rules exp rewrite-rule-name protocol]</code> hierarchy level. The IP precedence rewrite rules explanation does not apply to ACX Series Universal Access routers.</p> <p>On interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series 3D Universal Edge Routers and EX Series switches, we highly recommend that you configure the <b>default</b> option when you configure a behavior aggregate (BA) classifier that does not include a specific rewrite rule for MPLS packets. Doing so ensures that MPLS exp value is rewritten according to the BA classifier rules configured for forwarding or packet loss priority. This does not apply to ACX Series Universal Access routers.</p> <p><b>protocol-types</b>—Specify one or more protocol matching criteria:</p> <ul style="list-style-type: none"> <li>• <b>mpls-any</b>—Apply to MPLS packets, write MPLS header only.</li> <li>• <b>mpls-inet-both</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header.</li> <li>• <b>mpls-inet-both-non-vpn</b>—Apply to IPv4 MPLS packets, write MPLS and IPv4 header for only non VPN traffic.</li> </ul>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</a></li> <li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li> </ul>

- [rewrite-rules \(Definition\) on page 504](#)

## exp-push-push-push

---

<b>Syntax</b>	exp-push-push-push default;
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	For M Series routers, rewrite the EXP bits of all three labels of an outgoing packet, thereby maintaining CoS of an incoming non-MPLS packet.
<b>Options</b>	<b>default</b> —Apply the default MPLS EXP rewrite table.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</i></li> <li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li> <li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li> <li>• <a href="#">exp on page 377</a></li> <li>• <a href="#">exp-swap-push-push on page 488</a></li> <li>• <a href="#">ieee-802.1 (Rewrite Rules on Logical Interface) on page 379</a></li> <li>• <a href="#">ieee-802.1ad on page 381</a></li> <li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## exp-swap-push-push

---

<b>Syntax</b>	exp-swap-push-push default;
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	For M Series routers, rewrite the EXP bits of all three labels of an outgoing packet, thereby maintaining CoS of an incoming MPLS packet.
<b>Options</b>	<b>default</b> —Apply the default MPLS EXP rewrite table.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <i>Rewriting the EXP Bits of All Three Labels of an Outgoing Packet</i></li><li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li><li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li><li>• <a href="#">exp on page 377</a></li><li>• <a href="#">exp-push-push-push on page 487</a></li><li>• <a href="#">ieee-802.1 (Rewrite Rules on Logical Interface) on page 379</a></li><li>• <a href="#">ieee-802.1ad on page 381</a></li><li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## forwarding-class (BA Classifiers)

---

<b>Syntax</b>	<code>forwarding-class <i>class-name</i> {     <code>loss-priority level</code> <code>code-points</code> [ <i>aliases</i> ] [ <i>bit-patterns</i> ]; }</code>
<b>Hierarchy Level</b>	[edit class-of-service classifiers <i>type classifier-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Define forwarding class name and option values.
<b>Options</b>	<i>class-name</i> —Name of the forwarding class.  The remaining statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Behavior Aggregate Classifiers on page 36</a></li></ul>

## ieee-802.1 (Rewrite Rules on Logical Interface)

---

<b>Syntax</b>	ieee-802.1 ( <i>rewrite-name</i>   default) <b>vlan-tag</b> (outer   outer-and-inner);
<b>Hierarchy Level</b>	[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> <b>rewrite-rules</b> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. <b>vlan-tag</b> statement introduced in Junos OS Release 8.1.
<b>Description</b>	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.
<b>Options</b>	<b>rewrite-name</b> —Name of a <b>rewrite-rules</b> mapping configured at the [edit class-of-service <b>rewrite-rules</b> <b>ieee-802.1</b> ] hierarchy level.  <b>default</b> —The default mapping.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <a href="#">dscp (Rewrite Rules) on page 396</a></li><li>• <a href="#">dscp-ipv6 (CoS Rewrite Rules) on page 376</a></li><li>• <a href="#">exp on page 377</a></li><li>• <a href="#">exp-push-push-push on page 487</a></li><li>• <a href="#">exp-swap-push-push on page 488</a></li><li>• <a href="#">ieee-802.1ad on page 381</a></li><li>• <a href="#">inet-precedence (CoS Rewrite Rules) on page 382</a></li><li>• <a href="#">rewrite-rules (Definition) on page 504</a></li></ul>

## import (Rewrite Rules)

<b>Syntax</b>	<code>import (rewrite-name   default);</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service rewrite-rules type rewrite-name]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Specify a default or previously defined <b>rewrite-rules</b> mapping to import.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a rewrite-rules mapping configured at the <code>[edit class-of-service rewrite-rules]</code> hierarchy level.</p> <p><b>default</b>—The default <b>rewrite-rules</b> mapping.</p>
<b>Required Privilege Level</b>	<p><b>interface</b>—To view this statement in the configuration.</p> <p><b>interface-control</b>—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## inet-precedence (CoS Rewrite Rules)

<b>Syntax</b>	<code>inet-precedence (rewrite-name   default);</code>
<b>Hierarchy Level</b>	<code>[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules]</code>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Apply a IPv4 precedence rewrite rule.
<b>Options</b>	<p><b>rewrite-name</b>—Name of a <b>rewrite-rules</b> mapping configured at the <code>[edit class-of-service rewrite-rules inet-precedence]</code> hierarchy level.</p> <p><b>default</b>—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.</p>
<b>Required Privilege Level</b>	<p><b>interface</b>—To view this statement in the configuration.</p> <p><b>interface-control</b>—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> <li>• <a href="#">Applying Rewrite Rules to Output Logical Interfaces</a></li> <li>• <a href="#">protocol (Rewrite Rules) on page 502</a></li> <li>• <a href="#">rewrite-rules (Definition) on page 504</a></li> </ul>

## interfaces

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

```


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

```

<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Interface-set level added in Junos OS Release 8.5.
<b>Description</b>	Configure interface-specific CoS properties for incoming packets.
<b>Options</b>	The remaining statements are explained separately.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Overview of BA Classifier Types</i></li> <li>• <a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## loss-priority (BA Classifiers)

---

<b>Syntax</b>	<code>loss-priority <i>level</i>;</code>
<b>Hierarchy Level</b>	[edit class-of-service classifiers <i>type classifier-name</i> forwarding-class <i>class-name</i> ]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.
<b>Description</b>	Specify packet loss priority value for a specific set of code-point aliases and bit patterns.
<b>Options</b>	<p><i>level</i> can be one of the following:</p> <ul style="list-style-type: none"><li>• <b>high</b>—Packet has high loss priority.</li><li>• <b>medium-high</b>—Packet has medium-high loss priority.</li><li>• <b>medium-low</b>—Packet has medium-low loss priority.</li><li>• <b>low</b>—Packet has low loss priority.</li></ul>
<div> <b>NOTE:</b> <b>medium-low</b> priority is not supported on PTX1000 routers.</div>	
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic on page 26</a></li><li>• <a href="#">Configuring Tricolor Marking on page 92</a></li></ul>

## protocol (Rewrite Rules)

<b>Syntax</b>	<code>protocol protocol-types;</code>
<b>Hierarchy Level</b>	<p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules exp <i>rewrite-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules dscp <i>rewrite-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules dscp-ipv6 <i>rewrite-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i> rewrite-rules inet-prec <i>rewrite-name</i>]</p>
<b>Release Information</b>	<p>Statement introduced before Junos OS Release 7.4.</p> <p>Option for <b>dscp</b> and <b>inet-prec</b> introduced in Junos OS Release 8.4.</p> <p>Option for <b>dscp-ipv6</b> introduced in Junos OS Release 10.4R2.</p>
<b>Description</b>	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 packets, and write the CoS value to MPLS and IPv4 headers.
<b>Options</b>	<p><b>protocol-types</b> can be one of the following:</p> <ul style="list-style-type: none"> <li>• <b>mpls</b>—Apply a rewrite rule to MPLS packets and write the CoS value to MPLS headers.</li> <li>• <b>mpls-inet-both</b>—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.</li> <li>• <b>mpls-inet-both-non-vpn</b>—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.</li> </ul>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <i>Rewriting MPLS and IPv4 Packet Headers</i></li> </ul>

## rewrite-rules (Definition)

---

<b>Syntax</b>	<pre>rewrite-rules {     type <i>rewrite-name</i>{         import (<i>rewrite-name</i>   default);         forwarding-class <i>class-name</i> {             loss-priority <i>level</i> <i>code-point</i> [ <i>aliases</i> ] [ <i>bit-patterns</i> ];         }     } }</pre>
<b>Hierarchy Level</b>	[edit class-of-service]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4. ieee-802.1ad option introduced in Junos OS Release 9.2.
<b>Description</b>	Specify a rewrite-rules mapping for the traffic that passes through all queues on the interface.
<b>Options</b>	<p><i>rewrite-name</i>—Name of a rewrite-rules mapping.</p> <p><i>type</i>—Traffic type.</p> <p><b>Values:</b> dscp, dscp-ipv6, exp, ieee-802.1, ieee-802.1ad, inet-precedence</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"><li>• <a href="#">Configuring Rewrite Rules on page 215</a></li><li>• <i>Example: Configuring CoS for a PBB Network on MX Series Routers</i></li><li>• J Series router documentation</li></ul>

## rewrite-rules (Interfaces)

<b>Syntax</b>	<pre>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);   ieee-802.1ad (rewrite-name   default) vlan-tag (outer   outer-and-inner);   inet-precedence (rewrite-name   default); }</pre>
<b>Hierarchy Level</b>	<p>[edit class-of-service interfaces <i>interface-name</i>],</p> <p>[edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i>]</p>
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	<p>Associate a rewrite-rules configuration or default mapping with a specific interface.</p> <p>The [edit class-of-service interfaces <i>interface-name</i>] hierarchy level is not supported on M Series routers.</p> <p>The [edit class-of-service interfaces <i>interface-name</i> unit <i>logical-unit-number</i>] hierarchy level is not supported on ACX Series routers.</p> <p>On MX Series routers, although you can configure firewall filters and CoS rewrite rules on IRB interfaces, we recommend that you do not configure these functionalities on IRB interfaces because they do not work properly.</p> <p>On an MX Series router and on an EX Series switch, <b>exp-push-push-push</b>, <b>exp-swap-push-push</b>, and <b>frame-relay-de</b> are not supported on an integrated routing and bridging (IRB) interface.</p> <p>On an ACX Series router, only the outer tag is supported for <b>dscp</b>, <b>inet-precedence</b>, and <b>ieee802.1</b>.</p> <p>On M Series routers only, if you include the <b>control-word</b> statement at the [edit protocols l2circuit neighbor address interface <i>interface-name</i>] hierarchy level, the software cannot rewrite MPLS EXP bits.</p> <p>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.</p> <p>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.</p>

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.

**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 215](#)
- [rewrite-rules \(Definition\) on page 504](#)
- *Applying Rewrite Rules to Output Logical Interfaces*

## unit

<b>Syntax</b>	<pre> unit logical-unit-number {   classifiers {     type (classifier-name   default) family (mpls   all);   }   forwarding-class class-name;   fragmentation-map map-name;   input-traffic-control-profile profiler-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; } </pre>
<b>Hierarchy Level</b>	[edit class-of-service <a href="#">interfaces</a> interface-name]
<b>Release Information</b>	Statement introduced before Junos OS Release 7.4.
<b>Description</b>	Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.
<b>Options</b>	<p><b>logical-unit-number</b>—Number of the logical unit.</p> <p><b>Range:</b> 0 through 16,384</p> <p>The remaining statements are explained separately.</p>
<b>Required Privilege Level</b>	<p>interface—To view this statement in the configuration.</p> <p>interface-control—To add this statement to the configuration.</p>
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>Overview of BA Classifier Types</li> <li><a href="#">Configuring Rewrite Rules on page 215</a></li> </ul>

## [edit interfaces] Hierarchy Level

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

```

interfaces {
  interface-name {
    ... the "interface-name" subhierarchy appears after the main [edit interfaces] hierarchy level ...
  }
}

```

```
}
interface-set interface-set-name {
  interface interface-name {
    (unit unit-number | vlan-tags-outer vlan-tag);
  }
}
irb {
  accounting-profile name;
  description text;

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

  traceoptions {
    flag flag;
  }
  (traps | no-traps);
  unit logical-unit-number {
    accounting-profile name;
    bandwidth rate;
    description text;
    disable;
    encapsulation type;
    family inet {
      accounting {
        destination-class-usage;
        source-class-usage {
          input;
          output;
        }
      }
    }
  }
  address ipv4-address {
    arp ip-address (mac | multicast-mac) mac-address <publish>;
    broadcast address;
    preferred;
    primary;
    vrrp-group group-id {
      (accept-data | no-accept-data);
      advertise-interval seconds;
      advertisements-threshold number;
      authentication-key key;
      authentication-type authentication;
      fast-interval milliseconds;
      (preempt | no-preempt) {
        hold-time seconds;
      }
      priority number;
      track {
        interface interface-name {
          bandwidth-threshold bits-per-second priority-cost priority;
          priority-cost priority;
        }
        priority-hold-time seconds;
      }
    }
  }
}
```

```

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

```

```
    }
    virtual-inet6-address [addresses];
    virtual-link-local-address ipv6-address;
    vrrp-inherit-from {
        active-group group-number;
        active-interface interface-name;
    }
}
}
(dad-disable | no-dad-disable);
filter {
    input filter-name;
    output filter-name;
}
mtu bytes;
nd6-stale-time seconds;
no-neighbor-learn;
no-redirects;
policer {
    input policer-name;
    output policer-name;
}
rpf-check {
    fail-filter filter-name;
    mode {
        loose;
    }
}
}
}
family iso {
    address interface-address;
    mtu bytes;
}
family mpls {
    filter {
        input filter-name;
        output filter-name;
    }
    mtu bytes;
    policer {
        input policer-name;
        output policer-name;
    }
}
native-inner-vlan-id vlan-id;
proxy-arp (restricted | unrestricted);
(traps | no-traps);
vlan-id-list [vlan-id's];
vlan-id-range [vlan-id-range];
}
}
traceoptions {
    file <filename> <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
    flag flag <disable>;
    no-remote-trace;
```

```

    }
  }

interfaces {
  interface-name {
    disable;
    accounting-profile name;
    aggregated-ether-options {
      ethernet-switch-profile {
        tag-protocol-id [ hexadecimal-identifiers ];
      }
      (flow-control | no-flow-control);
      lacp {
        (active | passive);
        admin-key key;
        fast-failover;
        link-protection {
          disable;
          (revertive | non-revertive);
        }
        periodic (fast | slow);
        system-id mac-address;
        system-priority priority;
      }
      (link-protection | no-link-protection);
      link-speed (100m | 1g | 8g | 10g | 40g | 50g | 80g | 100g | oc192);
      logical-interface-fpc-redundancy;
      (loopback | no-loopback);
      mc-ae {
        chassis-id chassis-id;
        events {
          iccp-peer-down {
            force-icl-down;
            prefer-status-control-active;
          }
        }
        mc-ae-id mc-ae-id;
        mode (active-active | active-standby);
        redundancy-group group-id;
        status-control (active | standby);
      }
      minimum-links number;
      rebalance-periodic {
        start-time time;
        interval number;
      }
      source-address-filter {
        mac-address;
      }
      (source-filtering | no-source-filtering);
    }
  }
  auto-configure {
    remove-when-no-subscribers;
    stacked-vlan-ranges {
      access-profile profile-name;
      authentication {

```

```

        password password-string;
        username-include {
            circuit-type;
            delimiter delimiter-character;
            domain-name domain-name-string;
            interface-name;
            mac-address;
            option-82 ( circuit-id | remote-id);
            radius-realm radius-realm-string;
            user-prefix user-prefix-string;
        }
    }
    dynamic-profile profile-name {
        accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
        ranges (any | low-tag-high-tag), (any | low-tag-high-tag);
    }
}
vlan-ranges {
    access-profile profile-name;
    authentication {
        password password-string;
        username-include {
            circuit-type;
            delimiter delimiter-character;
            domain-name domain-name-string;
            interface-name;
            mac-address;
            option-82;
            radius-realm radius-realm-string;
            user-prefix user-prefix-string;
        }
    }
    dynamic-profile profile-name {
        accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
        ranges (any | low-tag)—(any | high-tag);
    }
}
override tag vlan-tag dynamic-profile profile name;
}
encapsulation (ethernet-bridge | ethernet-vpls | extended-vlan-bridge |
    extended-vlan-vpls | flexible-ethernet-services | vlan-vpls);
ether-options {
    802.3ad {
        aex;
        (backup | primary);
        lacp {
            force-up;
            port-priority
        }
    }
}
asynchronous-notification;
(auto-negotiation | no-auto-negotiation);
ethernet-switch-profile {
    ethernet-policer-profile {
        input-priority-map {
            ieee802.1p premium [ values ];

```

```

    }
    output-priority-map {
        classifier {
            premium {
                forwarding-class class-name {
                    loss-priority (high | low);
                }
            }
        }
    }
    policer cos-policer-name {
        aggregate {
            bandwidth-limit bps;
            burst-size-limit bytes;
        }
        premium {
            bandwidth-limit bps;
            burst-size-limit bytes;
        }
    }
    tag-protocol-id;
}
(mac-learn-enable | no-mac-learn-enable);
}
(flow-control | no-flow-control);
ignore-l3-incompletes;
link-mode (automatic | full-duplex | half-duplex);
(lloopback | no-loopback);
keepalives <interval seconds> <down-count number> <up-count number>;
speed (1g | 10m | 100m | 10m-100m | auto-negotiation);
source-address-filter {
    mac-address;
}
source-filtering | no-source-filtering;
}
flexible-vlan-tagging;
(gratuitous-arp-reply | no-gratuitous-arp-reply);
hold-time (up milliseconds | down milliseconds);
interface-transmit-statistics;
(keepalives <down-count number> <interval seconds> <up-count number> |
no-keepalives);
layer2-policer {
    apply-groups [ group-names ];
    apply-groups-except [ group-names ];
}
link-mode (automatic | full-duplex);
mac mac-address;
mtu bytes;
multi-chassis-protection peer-ip-address {
    interface interface-name;
}
native-vlan-id number;
no-gratuitous-arp-request;
optics-options {
    alarm low-light-alarm {
        (link-down | syslog);
    }
}

```

```

    }
    warning low-light-warning {
        (link-down | syslog);
    }
    wavelength nm;
}
passive-monitor-mode;
per-unit-scheduler;
speed (10m | 100m | 1g | auto | oc3 | oc12 | oc48);
stacked-vlan-tagging;
traceoptions {
    flag flag;
}
transmit-bucket {
    overflow discard;
    rate percentage;
    threshold bytes;
}
(traps | no-traps);
unidirectional;
vlan-tagging;
}

```

```

interface-name {
    unit logical-unit-number {
        disable;
        accept-source-mac {
            mac-address mac-address {
                policer {
                    input policer-name;
                    output policer-name;
                }
            }
        }
        accounting-profile name;
        advisory-options {
            downstream-rate rate;
            upstream-rate rate;
        }
        arp-resp (restricted|unrestricted);
        bandwidth rate;
        clear-dont-fragment-bit;
        copy-tos-to-outer-ip-header;
        demux-destination family;
        encapsulation (vlan-bridge | vlan-vpls);
        epd-threshold cells plp1 cells;
        filter filter-name;
        inner-vlan-id-range start start-id end end-id;
        input-vlan-map {
            (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
            inner-tag-protocol-id tpid;
            inner-vlan-id number;
            tag-protocol-id tpid;
            vlan-id number;
        }
    }
}

```

```

interface-shared-with psdnumerical-index;
layer2-policer {
    input-hierarchical-policer policer-name;
    input-policer policer-name;
    input-three-color policer-name;
    output-policer policer-name;
    output-three-color policer-name;
}
multi-chassis-protection peer-ip-address {
    interface interface-name;
}
native-inner-vlan-id number;
output-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
    inner-tag-protocol-id tpid;
    inner-vlan-id number;
    tag-protocol-id tpid;
    vlan-id number;
}
peer-interface interface-name;
peer-unit unit-number;
plp-to-clp;
proxy-arp <restricted | unrestricted>;
rpm {
    (client | server);
    twamp-server;
}
swap-by-poppush;
vlan-id number;
vlan-id-list [ vlan-id vlan-id-vlan-id ];
vlan-id-range number-number;
vlan-tags (inner <tpid.>vlan-id | inner-list [ vlan-id vlan-id-vlan-id ] |
    inner-range <tpid.>vlan-id-vlan-id) outer <tpid.>vlan-id;
}

unit logical-unit-number {
    family ethernet-switching {
        filter{
            group filter-group-number;
            (input filter-name | input-list [ filter-names ]);
            (output filter-name | output-list [ filter-names ]);
            (inner-vlan-id-list [ vlan-ids ] | vlan-id number | vlan-id-list [ number
                number-number ]);
            interface-mode (access | trunk);
            policer {
                input policer-name;
                output policer-name;
            }
            vlan-rewrite {
                translate old-vlan-id new-vlan-id;
            }
            vlan {
                members [ all vlan-identifiers ];
            }
        }
    }
    family inet {

```

```

filter {
    group filter-group-number;
    (input filter-name | input-list [ filter-names ]);
    (output filter-name | output-list [ filter-names ]);
}
input-hierarchical-policer policer-name;
mac-validate (loose | strict);
mtu bytes;
no-neighbor-learn;
no-redirects;
policer {
    arp policer-template-name;
    input policer-name;
    output policer-name;
}
primary;
receive-options-packets;
receive-ttl-exceeded;
rpf-check {
    fail-filter filter-name;
    mode loose;
}
sampling {
    (input | output | input output);
}
simple-filter {
    input filter-name;
}
targeted-broadcast {
    forward-and-send-to-re;
    forward-only;
}
unnumbered-address interface-name <destination address>
    <destination-profile profile-name> <preferred-source-address address>;
}

family inet6 {
    address ipv6-address {
        destination destination-address;
        eui-64;
        ndp ipv6-address <l2-interface interface-name> <(mac mac-address |
            multicast-mac multicast-mac-address) <publish>>;
        preferred;
        primary;
        vrrp-inet6-group group-number {
            (accept-data | no-accept-data);
            fast-interval milliseconds;
            inet6-advertise-interval seconds;
            (no-preempt; | ... the following preempt statement ...)
            preempt {
                hold-time seconds;
            }
            priority number;
            track {
                interface interface-name {

```

```

        bandwidth-threshold bits-per-second priority-cost priority;
        priority-cost priority;
    }
    priority-hold-time seconds;
    route ip-address-prefix/prefix-length routing-instance instance-name
        priority-cost priority;
    }
    virtual-inet6-address [ addresses ];
    virtual-link-local-address ipv6-address;
    vrrp-inherit-from {
        active-group group-number;
        active-interface interface-name;
    }
}
(dad-disable | no-dad-disable);
filter {
    group filter-group-number;
    (input filter-name | input-list [ filter-names ]);
    (output filter-name | output-list [ filter-names ]);
}
input-hierarchical-policer policer-name;
mtu bytes;
nd6-stale-time seconds;
no-neighbor-learn;
policer {
    input policer-name;
    output policer-name;
}
rpf-check {
    fail-filter filter-name;
    mode loose;
}
sampling {
    (input | output | input output);
}
unnumbered-address interface-name preferred-source-address address;
}

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

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

family mpls {
    filter {
        group filter-group-number;
        (input filter-name | input-list [ filter-names ]);
        (output filter-name | output-list [ filter-names ]);
    }
}

```

```
    }
    input-hierarchical-policer policer-name;
    maximum-labels maximum-labels;
    mtu bytes;
    policer {
        input policer-name;
        output policer-name;
    }
}

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

**Related Documentation**

- *Notational Conventions Used in Junos OS Configuration Hierarchies*

## copy-tos-to-outer-ip-header

<b>Syntax</b>	copy-tos-to-outer-ip-header;
<b>Hierarchy Level</b>	[edit interfaces <i>gr-fpc/pic/port</i> unit <i>logical-unit-number</i> ], [edit interfaces <i>gre</i> unit <i>logical-unit-number</i> ], [edit logical-systems <i>logical-system-name</i> interfaces <i>gr-fpc/pic/port</i> unit <i>logical-unit-number</i> ], [edit logical-systems <i>logical-system-name</i> interfaces <i>gre</i> unit <i>logical-unit-number</i> ]
<b>Release Information</b>	Statement introduced in Junos OS Release 8.2. Support for GRE interfaces for Generalized MPLS (GMPLS) introduced in Junos OS Release 12.3R7. Statement introduced in Junos OS Release 15.1X53-D10 for QFX10000 switches.
<b>Description</b>	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.
<b>Default</b>	If you omit this statement, the ToS bits in the outer IP header are set to 0.
<b>Required Privilege Level</b>	interface—To view this statement in the configuration. interface-control—To add this statement to the configuration.
<b>Related Documentation</b>	<ul style="list-style-type: none"> <li>• <a href="#">Example: Configuring a GRE Tunnel to Copy ToS Bits to the Outer IP Header on page 252</a></li> </ul>

