

VYATTA - A BROCADE COMPANY

Vyatta System

QoS

REFERENCE GUIDE

QoS



Vyatta - A Brocade Company  
Suite 200  
1301 Shoreway Road  
Belmont, CA 94002  
[vyatta.com](http://vyatta.com)  
650 413 7200  
1 888 VYATTA 1 (US and Canada)

## **COPYRIGHT**

Copyright © 2005–2013 Vyatta, a Brocade company. All rights reserved.

Vyatta - A Brocade Company reserves the right to make changes to software, hardware, and documentation without notice. For the most recent version of documentation, visit the Vyatta - A Brocade Company web site at [vyatta.com](http://vyatta.com).

## **PROPRIETARY NOTICES**

Vyatta - A Brocade Company is a registered trademark of Brocade, Inc.

Hyper-V is a registered trademark of Microsoft Corporation.

VMware, VMware ESX, and VMware server are trademarks of VMware, Inc.

XenServer, and XenCenter are trademarks of Citrix Systems, Inc.

All other trademarks are the property of their respective owners.

RELEASE DATE: April 2013

DOCUMENT REVISION. 6.6R0S6 d02

RELEASED WITH: Brocade Vyatta 6.6R0S6

PART NO. 53-1002786-01

# Contents

<b>Quick List of Commands</b> .....	<b>viii</b>
<b>List of Examples</b> .....	<b>xiii</b>
<b>Preface</b> .....	<b>xiv</b>
Intended Audience .....	xv
Product Applicability .....	xv
Organization of This Guide .....	xv
Document Conventions .....	xvi
Vyatta - A Brocade Company Publications .....	xvii
<b>Chapter 1 QoS Overview</b> .....	<b>1</b>
QoS Mechanisms .....	2
Default Traffic Prioritization .....	2
Traffic Policies .....	2
Drop-Tail .....	3
Fair Queue .....	3
Round-Robin .....	3
Traffic Shaper .....	4
Rate Control .....	4
Random Detect .....	4
Network Emulator .....	4
Priority Queue .....	5
Traffic Limiter .....	5
<b>Chapter 2 QoS Configuration Examples</b> .....	<b>6</b>
Outbound Example - Traffic Shaping .....	7
Inbound Example - Traffic Limiting .....	10
Inbound Example - Controlling Bandwidth Across Multiple Interfaces .....	11
<b>Chapter 3 QoS Commands</b> .....	<b>12</b>
interfaces <interface> traffic-policy .....	22
show incoming .....	26
show queueing .....	27
traffic-policy drop-tail <policy-name> .....	29
traffic-policy drop-tail <policy-name> description <desc> .....	30
traffic-policy drop-tail <policy-name> queue-limit <limit> .....	31

traffic-policy fair-queue <policy-name> . . . . .	33
traffic-policy fair-queue <policy-name> description <desc> . . . . .	35
traffic-policy fair-queue <policy-name> hash-interval <seconds> . . . . .	36
traffic-policy fair-queue <policy-name> queue-limit <limit> . . . . .	38
traffic-policy limiter <policy-name> . . . . .	40
traffic-policy limiter <policy-name> class <class> . . . . .	42
traffic-policy limiter <policy-name> class <class> bandwidth . . . . .	44
traffic-policy limiter <policy-name> class <class> burst . . . . .	46
traffic-policy limiter <policy-name> class <class> description <desc> . . . . .	48
traffic-policy limiter <policy-name> class <class> match <match-name> . . . . .	50
traffic-policy limiter <policy-name> class <class> match <match-name> description <desc> . . . . .	52
traffic-policy limiter <policy-name> class <class> match <match-name> ether destination <mac-addr> . . . . .	54
traffic-policy limiter <policy-name> class <class> match <match-name> ether protocol <num> . . . . .	56
traffic-policy limiter <policy-name> class <class> match <match-name> ether source <mac-addr> . . . . .	58
traffic-policy limiter <policy-name> class <class> match <match-name> ip destination . . . . .	60
traffic-policy limiter <policy-name> class <class> match <match-name> ip dscp <value> . . . . .	62
traffic-policy limiter <policy-name> class <class> match <match-name> ip protocol <proto> . . . . .	64
traffic-policy limiter <policy-name> class <class> match <match-name> ip source . . . . .	66
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 destination . . . . .	68
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 dscp <value> . . . . .	70
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 protocol <proto> . . . . .	72
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 source . . . . .	74
traffic-policy limiter <policy-name> class <class> match <match-name> mark <fwmark> . . . . .	76
traffic-policy limiter <policy-name> class <class> match <match-name> vif <vlan-id> . . . . .	78
traffic-policy limiter <policy-name> class <class> priority <priority> . . . . .	80
traffic-policy limiter <policy-name> default . . . . .	82
traffic-policy limiter <policy-name> default bandwidth . . . . .	84
traffic-policy limiter <policy-name> default burst . . . . .	86
traffic-policy limiter <policy-name> default priority <priority> . . . . .	88
traffic-policy limiter <policy-name> description <desc> . . . . .	90
traffic-policy network-emulator <policy-name> . . . . .	91
traffic-policy network-emulator <policy-name> bandwidth . . . . .	92
traffic-policy network-emulator <policy-name> burst . . . . .	94
traffic-policy network-emulator <policy-name> description <desc> . . . . .	96
traffic-policy network-emulator <policy-name> network-delay . . . . .	97
traffic-policy network-emulator <policy-name> packet-corruption <percent> . . . . .	99
traffic-policy network-emulator <policy-name> packet-loss <percent> . . . . .	101
traffic-policy network-emulator <policy-name> packet-reordering <percent> . . . . .	103
traffic-policy network-emulator <policy-name> queue-limit <limit> . . . . .	105
traffic-policy priority-queue <policy-name> . . . . .	107
traffic-policy priority-queue <policy-name> class <class> . . . . .	109

traffic-policy priority-queue <policy-name> class <class> description <desc> . . . . .	111
traffic-policy priority-queue <policy-name> class <class> match <match-name> . . . . .	113
traffic-policy priority-queue <policy-name> class <class> match <match-name> description <desc> . . . . .	115
traffic-policy priority-queue <policy-name> class <class> match <match-name> ether destination <mac-addr> . . . . .	117
traffic-policy priority-queue <policy-name> class <class> match <match-name> ether protocol <num> . . . . .	119
traffic-policy priority-queue <policy-name> class <class> match <match-name> ether source <mac-addr> . . . . .	121
traffic-policy priority-queue <policy-name> class <class> match <match-name> interface <interface> . . . . .	123
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip destination . . . . .	125
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip dscp <value> . . . . .	127
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip protocol <proto> . . . . .	129
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip source . . . . .	131
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 destination . . . . .	133
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 dscp <value> . . . . .	135
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 protocol <proto> . . . . .	137
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 source . . . . .	139
traffic-policy priority-queue <policy-name> class <class> match <match-name> mark <fwmark> . . . . .	141
traffic-policy priority-queue <policy-name> class <class> match <match-name> vif <vlan-id> . . . . .	143
traffic-policy priority-queue <policy-name> class <class> queue-limit <limit> . . . . .	145
traffic-policy priority-queue <policy-name> class <class> queue-type <type> . . . . .	147
traffic-policy priority-queue <policy-name> default . . . . .	149
traffic-policy priority-queue <policy-name> default queue-limit <limit> . . . . .	150
traffic-policy priority-queue <policy-name> default queue-type <type> . . . . .	152
traffic-policy priority-queue <policy-name> description <desc> . . . . .	154
traffic-policy random-detect <policy-name> . . . . .	156
traffic-policy random-detect <policy-name> bandwidth . . . . .	158
traffic-policy random-detect <policy-name> description <desc> . . . . .	160
traffic-policy random-detect <policy-name> precedence <precedence> . . . . .	161
traffic-policy rate-control <policy-name> . . . . .	164
traffic-policy rate-control <policy-name> bandwidth . . . . .	166
traffic-policy rate-control <policy-name> burst . . . . .	168
traffic-policy rate-control <policy-name> description <desc> . . . . .	170
traffic-policy rate-control <policy-name> latency . . . . .	171
traffic-policy round-robin <policy-name> . . . . .	173
traffic-policy round-robin <policy-name> class <class> . . . . .	175
traffic-policy round-robin <policy-name> class <class> description <desc> . . . . .	177
traffic-policy round-robin <policy-name> class <class> match <match-name> . . . . .	179
traffic-policy round-robin <policy-name> class <class> match <match-name> description <desc> . . . . .	181
traffic-policy round-robin <policy-name> class <class> match <match-name> ether destination <mac-addr> . . . . .	183
traffic-policy round-robin <policy-name> class <class> match <match-name> ether protocol <num> . . . . .	185
traffic-policy round-robin <policy-name> class <class> match <match-name> ether source <mac-addr> . . . . .	187

traffic-policy round-robin <policy-name> class <class> match <match-name> interface <interface> . . . . .	189
traffic-policy round-robin <policy-name> class <class> match <match-name> ip destination . . . . .	191
traffic-policy round-robin <policy-name> class <class> match <match-name> ip dscp <value> . . . . .	193
traffic-policy round-robin <policy-name> class <class> match <match-name> ip protocol <proto> . . . . .	195
traffic-policy round-robin <policy-name> class <class> match <match-name> ip source . . . . .	197
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 destination . . . . .	199
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 dscp <value> . . . . .	201
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 protocol <proto> . . . . .	203
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 source . . . . .	205
traffic-policy round-robin <policy-name> class <class> match <match-name> mark <fwmark> . . . . .	207
traffic-policy round-robin <policy-name> class <class> match <match-name> vif <vlan-id> . . . . .	209
traffic-policy round-robin <policy-name> class <class> quantum <packets> . . . . .	211
traffic-policy round-robin <policy-name> class <class> queue-limit <limit> . . . . .	213
traffic-policy round-robin <policy-name> class <class> queue-type <type> . . . . .	215
traffic-policy round-robin <policy-name> default . . . . .	217
traffic-policy round-robin <policy-name> default quantum <packets> . . . . .	218
traffic-policy round-robin <policy-name> default queue-limit <limit> . . . . .	220
traffic-policy round-robin <policy-name> default queue-type <type> . . . . .	222
traffic-policy round-robin <policy-name> description <desc> . . . . .	224
traffic-policy shaper <policy-name> . . . . .	226
traffic-policy shaper <policy-name> bandwidth . . . . .	228
traffic-policy shaper <policy-name> class <class> . . . . .	230
traffic-policy shaper <policy-name> class <class> bandwidth . . . . .	232
traffic-policy shaper <policy-name> class <class> burst . . . . .	234
traffic-policy shaper <policy-name> class <class> ceiling . . . . .	236
traffic-policy shaper <policy-name> class <class> description <desc> . . . . .	238
traffic-policy shaper <policy-name> class <class> match <match-name> . . . . .	240
traffic-policy shaper <policy-name> class <class> match <match-name> description <desc> . . . . .	242
traffic-policy shaper <policy-name> class <class> match <match-name> ether destination <mac-addr> . . . . .	244
traffic-policy shaper <policy-name> class <class> match <match-name> ether protocol <num> . . . . .	246
traffic-policy shaper <policy-name> class <class> match <match-name> ether source <mac-addr> . . . . .	248
traffic-policy shaper <policy-name> class <class> match <match-name> interface <interface> . . . . .	250
traffic-policy shaper <policy-name> class <class> match <match-name> ip destination . . . . .	252
traffic-policy shaper <policy-name> class <class> match <match-name> ip dscp <value> . . . . .	254
traffic-policy shaper <policy-name> class <class> match <match-name> ip protocol <proto> . . . . .	256
traffic-policy shaper <policy-name> class <class> match <match-name> ip source . . . . .	258
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 destination . . . . .	260
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 dscp <value> . . . . .	262
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 protocol <proto> . . . . .	264
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 source . . . . .	266
traffic-policy shaper <policy-name> class <class> match <match-name> mark <fwmark> . . . . .	268

---

traffic-policy shaper <policy-name> class <class> match <match-name> vif <vlan-id> . . . . .	270
traffic-policy shaper <policy-name> class <class> priority <priority> . . . . .	272
traffic-policy shaper <policy-name> class <class> queue-limit <limit> . . . . .	274
traffic-policy shaper <policy-name> class <class> queue-type <type> . . . . .	276
traffic-policy shaper <policy-name> class <class> set-dscp <value> . . . . .	278
traffic-policy shaper <policy-name> default . . . . .	280
traffic-policy shaper <policy-name> default bandwidth. . . . .	281
traffic-policy shaper <policy-name> default burst . . . . .	283
traffic-policy shaper <policy-name> default ceiling . . . . .	285
traffic-policy shaper <policy-name> default priority <priority>. . . . .	287
traffic-policy shaper <policy-name> default queue-limit <limit> . . . . .	289
traffic-policy shaper <policy-name> default queue-type <type>. . . . .	291
traffic-policy shaper <policy-name> default set-dscp <value> . . . . .	293
traffic-policy shaper <policy-name> description <desc> . . . . .	295
<b>Glossary of Acronyms . . . . .</b>	<b>297</b>

# Quick List of Commands

Use this list to help you quickly locate commands.

interfaces <interface> traffic-policy . . . . .	22
show incoming . . . . .	26
show queueing . . . . .	27
traffic-policy drop-tail <policy-name> description <desc> . . . . .	30
traffic-policy drop-tail <policy-name> queue-limit <limit> . . . . .	31
traffic-policy drop-tail <policy-name> . . . . .	29
traffic-policy fair-queue <policy-name> description <desc> . . . . .	35
traffic-policy fair-queue <policy-name> hash-interval <seconds> . . . . .	36
traffic-policy fair-queue <policy-name> queue-limit <limit> . . . . .	38
traffic-policy fair-queue <policy-name> . . . . .	33
traffic-policy limiter <policy-name> class <class> bandwidth . . . . .	44
traffic-policy limiter <policy-name> class <class> burst . . . . .	46
traffic-policy limiter <policy-name> class <class> description <desc> . . . . .	48
traffic-policy limiter <policy-name> class <class> match <match-name> description <desc> . . . . .	52
traffic-policy limiter <policy-name> class <class> match <match-name> ether destination <mac-addr> . . . . .	54
traffic-policy limiter <policy-name> class <class> match <match-name> ether protocol <num> . . . . .	56
traffic-policy limiter <policy-name> class <class> match <match-name> ether source <mac-addr> . . . . .	58
traffic-policy limiter <policy-name> class <class> match <match-name> ip destination . . . . .	60
traffic-policy limiter <policy-name> class <class> match <match-name> ip dscp <value> . . . . .	62
traffic-policy limiter <policy-name> class <class> match <match-name> ip protocol <proto> . . . . .	64
traffic-policy limiter <policy-name> class <class> match <match-name> ip source . . . . .	66
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 destination . . . . .	68
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 dscp <value> . . . . .	70
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 protocol <proto> . . . . .	72
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 source . . . . .	74
traffic-policy limiter <policy-name> class <class> match <match-name> mark <fwmark> . . . . .	76
traffic-policy limiter <policy-name> class <class> match <match-name> vif <vlan-id> . . . . .	78
traffic-policy limiter <policy-name> class <class> match <match-name> . . . . .	50
traffic-policy limiter <policy-name> class <class> priority <priority> . . . . .	80
traffic-policy limiter <policy-name> class <class> . . . . .	42
traffic-policy limiter <policy-name> default bandwidth . . . . .	84
traffic-policy limiter <policy-name> default burst . . . . .	86



traffic-policy limiter <policy-name> default priority <priority> . . . . .	88
traffic-policy limiter <policy-name> default . . . . .	82
traffic-policy limiter <policy-name> description <desc> . . . . .	90
traffic-policy limiter <policy-name> . . . . .	40
traffic-policy network-emulator <policy-name> bandwidth . . . . .	92
traffic-policy network-emulator <policy-name> burst . . . . .	94
traffic-policy network-emulator <policy-name> description <desc> . . . . .	96
traffic-policy network-emulator <policy-name> network-delay . . . . .	97
traffic-policy network-emulator <policy-name> packet-corruption <percent> . . . . .	99
traffic-policy network-emulator <policy-name> packet-loss <percent> . . . . .	101
traffic-policy network-emulator <policy-name> packet-reordering <percent> . . . . .	103
traffic-policy network-emulator <policy-name> queue-limit <limit> . . . . .	105
traffic-policy network-emulator <policy-name> . . . . .	91
traffic-policy priority-queue <policy-name> class <class> description <desc> . . . . .	111
traffic-policy priority-queue <policy-name> class <class> match <match-name> description <desc> . . . . .	115
traffic-policy priority-queue <policy-name> class <class> match <match-name> ether destination <mac-addr> . . . . .	117
traffic-policy priority-queue <policy-name> class <class> match <match-name> ether protocol <num> . . . . .	119
traffic-policy priority-queue <policy-name> class <class> match <match-name> ether source <mac-addr> . . . . .	121
traffic-policy priority-queue <policy-name> class <class> match <match-name> interface <interface> . . . . .	123
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip destination . . . . .	125
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip dscp <value> . . . . .	127
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip protocol <proto> . . . . .	129
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip source . . . . .	131
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 destination . . . . .	133
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 dscp <value> . . . . .	135
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 protocol <proto> . . . . .	137
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 source . . . . .	139
traffic-policy priority-queue <policy-name> class <class> match <match-name> mark <fwmark> . . . . .	141
traffic-policy priority-queue <policy-name> class <class> match <match-name> vif <vlan-id> . . . . .	143
traffic-policy priority-queue <policy-name> class <class> match <match-name> . . . . .	113
traffic-policy priority-queue <policy-name> class <class> queue-limit <limit> . . . . .	145
traffic-policy priority-queue <policy-name> class <class> queue-type <type> . . . . .	147
traffic-policy priority-queue <policy-name> class <class> . . . . .	109
traffic-policy priority-queue <policy-name> default queue-limit <limit> . . . . .	150
traffic-policy priority-queue <policy-name> default queue-type <type> . . . . .	152
traffic-policy priority-queue <policy-name> default . . . . .	149

traffic-policy priority-queue <policy-name> description <desc> . . . . .	154
traffic-policy priority-queue <policy-name> . . . . .	107
traffic-policy random-detect <policy-name> bandwidth . . . . .	158
traffic-policy random-detect <policy-name> description <desc> . . . . .	160
traffic-policy random-detect <policy-name> precedence <precedence> . . . . .	161
traffic-policy random-detect <policy-name> . . . . .	156
traffic-policy rate-control <policy-name> bandwidth . . . . .	166
traffic-policy rate-control <policy-name> burst . . . . .	168
traffic-policy rate-control <policy-name> description <desc> . . . . .	170
traffic-policy rate-control <policy-name> latency . . . . .	171
traffic-policy rate-control <policy-name> . . . . .	164
traffic-policy round-robin <policy-name> class <class> description <desc> . . . . .	177
traffic-policy round-robin <policy-name> class <class> match <match-name> description <desc> . . . . .	181
traffic-policy round-robin <policy-name> class <class> match <match-name> ether destination <mac-addr> . . .	183
traffic-policy round-robin <policy-name> class <class> match <match-name> ether protocol <num> . . . . .	185
traffic-policy round-robin <policy-name> class <class> match <match-name> ether source <mac-addr> . . . . .	187
traffic-policy round-robin <policy-name> class <class> match <match-name> interface <interface> . . . . .	189
traffic-policy round-robin <policy-name> class <class> match <match-name> ip destination . . . . .	191
traffic-policy round-robin <policy-name> class <class> match <match-name> ip dscp <value> . . . . .	193
traffic-policy round-robin <policy-name> class <class> match <match-name> ip protocol <proto> . . . . .	195
traffic-policy round-robin <policy-name> class <class> match <match-name> ip source . . . . .	197
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 destination . . . . .	199
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 dscp <value> . . . . .	201
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 protocol <proto> . . . . .	203
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 source . . . . .	205
traffic-policy round-robin <policy-name> class <class> match <match-name> mark <fwmark> . . . . .	207
traffic-policy round-robin <policy-name> class <class> match <match-name> vif <vlan-id> . . . . .	209
traffic-policy round-robin <policy-name> class <class> match <match-name> . . . . .	179
traffic-policy round-robin <policy-name> class <class> quantum <packets> . . . . .	211
traffic-policy round-robin <policy-name> class <class> queue-limit <limit> . . . . .	213
traffic-policy round-robin <policy-name> class <class> queue-type <type> . . . . .	215
traffic-policy round-robin <policy-name> class <class> . . . . .	175
traffic-policy round-robin <policy-name> default quantum <packets> . . . . .	218
traffic-policy round-robin <policy-name> default queue-limit <limit> . . . . .	220
traffic-policy round-robin <policy-name> default queue-type <type> . . . . .	222
traffic-policy round-robin <policy-name> default . . . . .	217

traffic-policy round-robin <policy-name> description <desc> . . . . .	224
traffic-policy round-robin <policy-name> . . . . .	173
traffic-policy shaper <policy-name> bandwidth . . . . .	228
traffic-policy shaper <policy-name> class <class> bandwidth . . . . .	232
traffic-policy shaper <policy-name> class <class> burst . . . . .	234
traffic-policy shaper <policy-name> class <class> ceiling . . . . .	236
traffic-policy shaper <policy-name> class <class> description <desc> . . . . .	238
traffic-policy shaper <policy-name> class <class> match <match-name> description <desc> . . . . .	242
traffic-policy shaper <policy-name> class <class> match <match-name> ether destination <mac-addr> . . . . .	244
traffic-policy shaper <policy-name> class <class> match <match-name> ether protocol <num> . . . . .	246
traffic-policy shaper <policy-name> class <class> match <match-name> ether source <mac-addr> . . . . .	248
traffic-policy shaper <policy-name> class <class> match <match-name> interface <interface> . . . . .	250
traffic-policy shaper <policy-name> class <class> match <match-name> ip destination . . . . .	252
traffic-policy shaper <policy-name> class <class> match <match-name> ip dscp <value> . . . . .	254
traffic-policy shaper <policy-name> class <class> match <match-name> ip protocol <proto> . . . . .	256
traffic-policy shaper <policy-name> class <class> match <match-name> ip source . . . . .	258
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 destination . . . . .	260
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 dscp <value> . . . . .	262
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 protocol <proto> . . . . .	264
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 source . . . . .	266
traffic-policy shaper <policy-name> class <class> match <match-name> mark <fwmark> . . . . .	268
traffic-policy shaper <policy-name> class <class> match <match-name> vif <vlan-id> . . . . .	270
traffic-policy shaper <policy-name> class <class> match <match-name> . . . . .	240
traffic-policy shaper <policy-name> class <class> priority <priority> . . . . .	272
traffic-policy shaper <policy-name> class <class> queue-limit <limit> . . . . .	274
traffic-policy shaper <policy-name> class <class> queue-type <type> . . . . .	276
traffic-policy shaper <policy-name> class <class> set-dscp <value> . . . . .	278
traffic-policy shaper <policy-name> class <class> . . . . .	230
traffic-policy shaper <policy-name> default bandwidth . . . . .	281
traffic-policy shaper <policy-name> default burst . . . . .	283
traffic-policy shaper <policy-name> default ceiling . . . . .	285
traffic-policy shaper <policy-name> default priority <priority> . . . . .	287
traffic-policy shaper <policy-name> default queue-limit <limit> . . . . .	289
traffic-policy shaper <policy-name> default queue-type <type> . . . . .	291
traffic-policy shaper <policy-name> default set-dscp <value> . . . . .	293
traffic-policy shaper <policy-name> default . . . . .	280

---

traffic-policy shaper <policy-name> description <desc> .....	295
traffic-policy shaper <policy-name> .....	226

# List of Examples

Use this list to help you locate examples you'd like to look at or try.

Example 2-1 Traffic shaping .....	8
Example 2-2 Traffic limiting .....	10
Example 2-3 Traffic limiting across multiple interfaces .....	11
Example 3-1 “show incoming”: Displaying all incoming packet actions. ....	26
Example 3-2 “show queueing”: Displaying all outgoing QoS policies. ....	27
Example 3-3 “show queueing ethernet eth0”: Displaying QoS policies on a specific interface. ....	28

# Preface

This document describes the quality of service (QoS) mechanisms available on the Vyatta system.

This preface provides information about using this guide. The following topics are presented

- [Intended Audience](#)
- [Product Applicability](#)
- [Organization of This Guide](#)
- [Document Conventions](#)
- [Vyatta - A Brocade Company Publications](#)

---

## Intended Audience

---

This guide is intended for experienced system and network administrators. Depending on the functionality to be used, readers should have specific knowledge in the following areas:

- Networking and data communications
- TCP/IP protocols
- General router configuration
- Routing protocols
- Network administration
- Network security
- IP services

## Product Applicability

This guide describes software for the following products:

- Brocade Vyatta 5410F vRouter
- Brocade Vyatta 5415F vRouter

---

## Organization of This Guide

---

This guide has the following aid to help you find the information you are looking for:

- [Quick List of Commands](#)

Use this list to help you quickly locate commands.

- [List of Examples](#)

Use this list to help you locate examples you'd like to try or look at.

This guide has the following chapters:

Chapter	Description	Page
<a href="#">Chapter 1: QoS Overview</a>	<a href="#">This chapter provides a brief overview of quality of service (QoS) features on the Vyatta system.</a>	<a href="#">1</a>

Chapter 2: QoS Configuration Examples	This chapter provides configuration examples for implementing quality of service (QoS) on the Vyatta system.	6
Chapter 3: QoS Commands	This chapter describes commands for QoS features supported by the Vyatta system.	12
Glossary of Acronyms		297

## Document Conventions

This guide uses the following advisory paragraphs, as follows.



**WARNING** Warnings alert you to situations that may pose a threat to personal safety.



**CAUTION** Cautions alert you to situations that might cause harm to your system or damage to equipment, or that may affect service.

**NOTE** Notes provide information you might need to avoid problems or configuration errors.

This document uses the following typographic conventions.

Monospace	Examples, command-line output, and representations of configuration nodes.
<b>bold Monospace</b>	Your input: something you type at a command line.
<b>bold</b>	Commands, keywords, and file names, when mentioned inline. Objects in the user interface, such as tabs, buttons, screens, and panes.
<i>italics</i>	An argument or variable where you supply a value.
<key>	A key on your keyboard, such as <Enter>. Combinations of keys are joined by plus signs (“+”), as in <Ctrl>+c.
[ key1   key2]	Enumerated options for completing a syntax. An example is [enable   disable].
<i>num1–numN</i>	A inclusive range of numbers. An example is 1–65535, which means 1 through 65535, inclusive.



---

<i>arg1..argN</i>	A range of enumerated values. An example is eth0..eth3, which means eth0, eth1, eth2, or eth3.
<i>arg[ arg...]</i> <i>arg[,arg...]</i>	A value that can optionally represent a list of elements (a space-separated list and a comma-separated list, respectively).

---

## Vyatta - A Brocade Company Publications

---

Full product documentation is provided in the Vyatta technical library. To see what documentation is available for your release, see the *Guide to Vyatta Documentation*. This guide is posted with every release of Vyatta software and provides a great starting point for finding the information you need.

Additional information is available on [www.vyatta.com](http://www.vyatta.com) and [www.vyatta.org](http://www.vyatta.org).

# Chapter 1: QoS Overview

This chapter provides a brief overview of quality of service (QoS) features on the Vyatta system.

This chapter presents the following topics:

- [QoS Mechanisms](#)
- [Default Traffic Prioritization](#)
- [Traffic Policies](#)

## QoS Mechanisms

---

Quality of Service (QoS) is a feature that allows network administrators to identify different traffic flows and then treat them according to their individual requirements, rather than simply using the default mechanism.

On the Vyatta system the default QoS mechanism is based on queue prioritization. In addition to the default queuing mechanism, the Vyatta system provides a variety of QoS mechanisms for identifying and treating the various traffic flows that pass through an interface. In general, these can be categorized as mechanisms that apply to outbound traffic and mechanisms that apply to inbound traffic.

The general workflow for non-default QoS mechanisms (traffic policies) is as follows:

- 1 Create a traffic policy (traffic-policy). The policy identifies traffic flows and specifies how each flow is to be treated.
- 2 Apply the policy to an interface.

## Default Traffic Prioritization

---

By default, all traffic sent out by the Vyatta system is prioritized based on the value in its Type of Service (ToS) field into one of three priority queues. The packets on the highest priority queue are sent out first, followed by those on the next-highest priority queue, followed by those on the lowest priority queue. Within each queue, all packets are sent on a First In First Out (FIFO) basis and receive “best effort” delivery. If traffic arrives on a queue faster than it can be delivered (for example, because of bandwidth limitations) it is buffered within the system. If more data arrives than the system can buffer, the excess is dropped.

Data traffic is divided in this way because providing equal levels of service for all traffic is not always desirable. Some types of traffic, by their nature, should be treated differently than others. For example, voice traffic is very sensitive to delay and, if it is not processed accordingly, may be unintelligible. Data, on the other hand, is sensitive not to delay but to corruption.

## Traffic Policies

---

The following table shows the traffic policies supported by the Vyatta system and whether they can be applied to inbound traffic or outbound traffic:

Traffic policy	Inbound	Outbound
Drop-Tail	-	Yes
Fair Queue	-	Yes
Round-Robin	-	Yes
Traffic Shaper	-	Yes
Rate Control	-	Yes
Random Detect	-	Yes
Network Emulator	-	Yes
Priority Queue	-	Yes
Traffic Limiter	Yes	-

## Drop-Tail

The “**traffic-policy drop-tail**” mechanism is a scheduling algorithm. It provides pure FIFO (First In First Out) queuing; in other words, data packets are transmitted in the same order that they arrive. If the queue fills up, then the “tail” of the queue (that is, the set of packets just arriving in the queue) is dropped. With drop-tail queuing, there is a single queue and all traffic is treated equally; traffic is not prioritized as it is in the default case.

## Fair Queue

The “**traffic-policy fair-queue**” mechanism is a scheduling algorithm. It provides queuing based on the Stochastic Fairness Queuing algorithm. In this queuing algorithm, traffic flows are identified by IP protocol, source address, and/or destination address. Flows thus identified receive fair access to network resources such that no one flow is permitted to use the majority of the bandwidth.

## Round-Robin

The “**traffic-policy round-robin**” mechanism is a simple scheduling algorithm. In round-robin queueing, classes of traffic are identified and bandwidth is divided equally among the defined classes.

## Traffic Shaper

The “**traffic-policy shaper**” mechanism provides queuing based on the Token Bucket shaping algorithm. This algorithm allows for bursting if a “bucket” has tokens to “spend.” The difference between the **shaper** and **round-robin** algorithms is that the **shaper** algorithm limits bandwidth usage by class and then allocates any leftover bandwidth. **Round-robin**, on the other hand, attempts to divide all available bandwidth equally between the defined classes.

## Rate Control

The “**traffic-policy rate-control**” mechanism is a scheduling algorithm. It provides queuing based on the Token Bucket Filter algorithm. This algorithm only passes packets arriving at a rate which does not exceed an administratively set rate. It is possible, however, for short bursts of traffic to occur in excess of this rate.

## Random Detect

The “**traffic-policy random-detect**” mechanism is a congestion avoidance mechanism that includes Random Early Detection (RED) and Weighted Random Early Detection (WRED).

Congestion occurs when output buffers are allowed to fill such that packets must be dropped. Congestion can cause global resynchronization of TCP hosts as multiple hosts reduce their transmission rates to try to clear the congestion; this can significantly affect network performance. As congestion clears, the network increases transmission rates again until the point where congestion reoccurs. This cycle of congestion and clearing does not make the best use of the available bandwidth.

RED reduces the chance that network congestion will occur by randomly dropping packets when the output interface begins to show signs of congestion. The packet-dropping acts as a signal to the source to decrease its transmission rate which, in turn, helps avoid conditions of congestion and reduces the chance of global synchronization, making better use of network bandwidth.

WRED takes RED one step further by providing a way to attach precedence to different traffic streams. Differential quality of service can then be provided to different traffic streams by dropping more packets from some streams than from others.

## Network Emulator

The “**traffic-policy network-emulator**” mechanism provides a way to emulate WAN traffic. It is typically used for system testing.

## Priority Queue

The “**traffic-policy priority-queue**” mechanism is scheduling algorithm. It provides up to seven queues, each with a different priority. Packets are placed in the queues based on match criteria associated with each queue. Packets are retrieved from the queues in priority order. Packets in lower priority queues will not be transmitted until those is higher priority queues have been sent. If packets continually fill higher priority queues, those waiting on lower priority queues will not be serviced until the higher priority traffic load abates.

## Traffic Limiter

The “**traffic-policy limiter**” mechanism can be used to throttle (or “police”) incoming traffic. The mechanism assigns each traffic flow a bandwidth limit. All incoming traffic within a flow in excess of the bandwidth limit is dropped. The advantages are that this policy does not incur queuing delay and it is the only policy that can be applied to inbound traffic. The disadvantage is that it is more likely to drop packets and cause retransmissions. **Shaper** or **rate-control** are typically used to throttle outgoing traffic where queuing delays can be tolerated. They will buffer traffic in excess of the bandwidth limit and will not drop packets unless the buffers overflow.

## Chapter 2: QoS Configuration Examples

This chapter provides configuration examples for implementing quality of service (QoS) on the Vyatta system.

This chapter presents the following topics:

- [Outbound Example - Traffic Shaping](#)
- [Inbound Example - Traffic Limiting](#)
- [Inbound Example - Controlling Bandwidth Across Multiple Interfaces](#)

## Outbound Example - Traffic Shaping

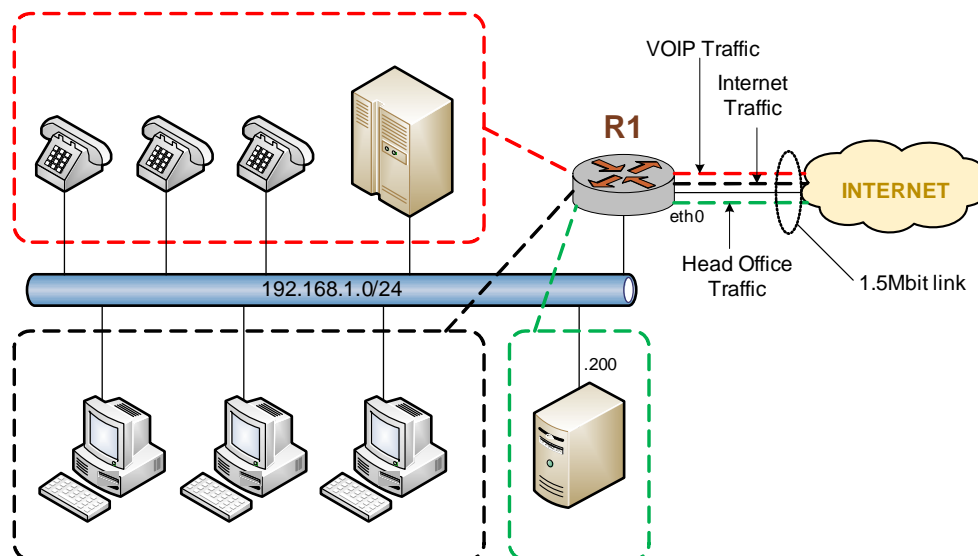
Figure 2-1 shows a simple site using QoS on the Vyatta system (R1) to treat three distinct traffic flows in different ways. This scenario represents a single branch office with a Voice over IP (VoIP) phone system, users that connect to the Internet, and a server that requires a relatively high-speed connection to head office. In this example:

- All traffic flows through a 1.5 Mbit link to the Internet Service Provider (ISP).
- A minimum 50% of this bandwidth is to be set aside for the VoIP traffic, 35% for the head office traffic, and 15% for all other traffic.
- All traffic flows will use available bandwidth beyond their minimum configured rates.
- In addition, the VoIP traffic is to be categorized into two distinct flows:
  - 5% of bandwidth is to be used for control traffic (in the example, Session Initiation Protocol [SIP] signals for setting up calls).
  - 45% of bandwidth is to be used for Real Time Protocol (RTP) media.

The different flows are identified by their Differentiated Services Code Point (DSCP) values: SIP traffic is assigned a DSCP value of 26 and RTP traffic is assigned a DSCP value of 46.)

- The head office traffic arrives from a single server at IP address 192.168.1.200.

Figure 2-1 Example VoIP site using QoS





To configure this scenario, perform the following steps in configuration mode.

#### Example 2-1 Traffic shaping

Step	Command
Create the configuration node for the QoS policy.	<code>vyatta@R1# set traffic-policy shaper OFFICE</code>
Add a description.	<code>vyatta@R1# set traffic-policy shaper OFFICE description "QoS policy for office WAN"</code>
Set the overall link bandwidth.	<code>vyatta@R1# set traffic-policy shaper OFFICE bandwidth 1500kbit</code>
Add a description for the first traffic class - VOIP data traffic.	<code>vyatta@R1# set traffic-policy shaper OFFICE class 10 description "VOIP - RTP traffic"</code>
Assign bandwidth to the VOIP data traffic.	<code>vyatta@R1# set traffic-policy shaper OFFICE class 10 bandwidth 45%</code>
Allow the VOIP data traffic to use all available bandwidth.	<code>vyatta@R1# set traffic-policy shaper OFFICE class 10 ceiling 100%</code>
Identify the VOIP data traffic (DSCP=46).	<code>vyatta@R1# set traffic-policy shaper OFFICE class 10 match VOIP-RTP ip dscp 46</code>
Add a description for the second traffic class - VOIP control traffic.	<code>vyatta@R1# set traffic-policy shaper OFFICE class 20 description "VOIP -SIP traffic"</code>
Assign bandwidth to the VOIP control traffic.	<code>vyatta@R1# set traffic-policy shaper OFFICE class 20 bandwidth 5%</code>
Allow the VOIP control traffic to use all available bandwidth.	<code>vyatta@R1# set traffic-policy shaper OFFICE class 20 ceiling 100%</code>
Identify the VOIP control traffic (DSCP=26).	<code>vyatta@R1# set traffic-policy shaper OFFICE class 20 match VOIP-SIP ip dscp 26</code>
Add a description for the third traffic class - head office traffic.	<code>vyatta@R1# set traffic-policy shaper OFFICE class 30 description "Head office traffic"</code>
Assign bandwidth to the head office traffic.	<code>vyatta@R1# set traffic-policy shaper OFFICE class 30 bandwidth 35%</code>
Allow the head office traffic to use all available bandwidth.	<code>vyatta@R1# set traffic-policy shaper OFFICE class 30 ceiling 100%</code>
Identify the head office traffic (IP address=192.168.1.200/24).	<code>vyatta@R1# set traffic-policy shaper OFFICE class 30 match HO-TRAFFIC ip source address 192.168.1.200/24</code>
Assign bandwidth to the remainder of the traffic.	<code>vyatta@R1# set traffic-policy shaper OFFICE default bandwidth 15%</code>

## Example 2-1 Traffic shaping

Allow the default traffic to use all available bandwidth.	vyatta@R1# set traffic-policy shaper OFFICE default ceiling 100%
Commit the change.	vyatta@R1# commit
Show the traffic-policy configuration.	<pre> vyatta@R1# show traffic-policy shaper OFFICE {     bandwidth 1500kbit     class 10 {         bandwidth 45%         ceiling 100%         description "VOIP - RTP traffic"         match VOIP-RTP {             ip {                 dscp 46             }         }     }     class 20 {         bandwidth 5%         ceiling 100%         description "VOIP - SIP traffic"         match VOIP-SIP {             ip {                 dscp 26             }         }     }     class 30 {         bandwidth 35%         ceiling 100%         description "Head office traffic"         match HO-TRAFFIC {             ip {                 source {                     address 192.168.1.200/24                 }             }         }     }     default {         bandwidth 15%         ceiling 100%     }     description "QoS policy for office WAN" } </pre>

**Example 2-1 Traffic shaping**

Assign the QoS policy to the upstream interface.	<code>vyatta@R1# set interfaces ethernet eth0 traffic-policy out OFFICE</code>
--	--

## Inbound Example - Traffic Limiting

In this example inbound mail traffic (port 25) on eth0 is limited to 300kbit/sec.

To configure this scenario, perform the following steps in configuration mode.

**Example 2-2 Traffic limiting**

Step	Command
Create the configuration node for the QoS policy.	<code>vyatta@R1# set traffic-policy limiter LIMIT-MAIL</code>
Add a description for the traffic class - Mail traffic.	<code>vyatta@R1# set traffic-policy limiter LIMIT-MAIL class 10 description "Limit inbound mail traffic"</code>
Assign bandwidth to the Mail data traffic.	<code>vyatta@R1# set traffic-policy limiter LIMIT-MAIL class 10 bandwidth 300kbit</code>
Identify the Mail data traffic (port=25).	<code>vyatta@R1# set traffic-policy limiter LIMIT-MAIL class 10 match MAIL-TRAFFIC ip destination port 25</code>
Commit the change.	<code>vyatta@R1# commit</code>
Show the traffic-policy configuration.	<code>vyatta@R1# show traffic-policy limiter LIMIT-MAIL {   class 10 {     bandwidth 300kbit     description "Limit inbound mail traffic"     match MAIL-TRAFFIC {       ip {         destination {           port 25         }       }     }   } }</code>
Assign the QoS policy to inbound traffic on eth0.	<code>vyatta@R1# set interfaces ethernet eth0 traffic-policy in LIMIT-MAIL</code>

## Inbound Example - Controlling Bandwidth Across Multiple Interfaces

In this example the combined inbound traffic from eth0, eth1, and eth2 is not to exceed 1Gbit/sec. To do this, inbound traffic from these interfaces is redirected to Input interface ifb0. A rate control policy to limit traffic to 1Gbit/sec is created and is assigned to ifb0.

To configure this scenario, perform the following steps in configuration mode.

Example 2-3 Traffic limiting across multiple interfaces

Step	Command
Redirect eth0 traffic to input interface ifb0.	<code>vyatta@R1# set interfaces ethernet eth0 redirect ifb0</code>
Redirect eth1 traffic to input interface ifb0.	<code>vyatta@R1# set interfaces ethernet eth1 redirect ifb0</code>
Redirect eth2 traffic to input interface ifb0.	<code>vyatta@R1# set interfaces ethernet eth2 redirect ifb0</code>
Create the configuration node for the QoS policy.	<code>vyatta@R1# set traffic-policy rate-control LIMIT-1Gbit</code>
Add a description for the policy.	<code>vyatta@R1# set traffic-policy rate-control LIMIT-1Gbit description "Limit traffic to 1Gbit"</code>
Assign a bandwidth limit to the traffic.	<code>vyatta@R1# set traffic-policy rate-control LIMIT-1Gbit bandwidth 1gbit</code>
Commit the change.	<code>vyatta@R1# commit</code>
Show the traffic-policy configuration.	<code>vyatta@R1# show traffic-policy rate-control LIMIT-1Gbit {     bandwidth 1gbit     description "Limit traffic to 1Gbit" }</code>
Assign the QoS policy to outbound traffic on ifb0 (which will be the combined traffic from eth0, eth1, and eth2). Outbound traffic from an input interface is internal to the Vyatta device.	<code>vyatta@R1# set interfaces input ifb0 traffic-policy out LIMIT-1Gbit</code>

## Chapter 3: QoS Commands

This chapter describes commands for QoS features supported by the Vyatta system.

This chapter contains the following commands.

Configuration Commands	
Applying QoS Policies to Interfaces	
<code>interfaces &lt;interface&gt; traffic-policy</code>	Applies a QoS policy to the specified interface.
Outbound - Drop Tail Policies	
<code>traffic-policy drop-tail &lt;policy-name&gt;</code>	Defines a drop tail (pure FIFO) QoS policy.
<code>traffic-policy drop-tail &lt;policy-name&gt; description &lt;desc&gt;</code>	Sets a description for a drop tail policy.
<code>traffic-policy drop-tail &lt;policy-name&gt; queue-limit &lt;limit&gt;</code>	Sets an upper bound for the number of packets allowed in the queue for a drop tail policy.
Outbound - Fair Queue Policies	
<code>traffic-policy fair-queue &lt;policy-name&gt;</code>	Defines a fair queue QoS policy.
<code>traffic-policy fair-queue &lt;policy-name&gt; description &lt;desc&gt;</code>	Sets a description for a fair queue policy.
<code>traffic-policy fair-queue &lt;policy-name&gt; hash-interval &lt;seconds&gt;</code>	Specifies the interval between flow hash function updates for a fair queue policy.
<code>traffic-policy fair-queue &lt;policy-name&gt; queue-limit &lt;limit&gt;</code>	Sets an upper bound for the number of packets allowed in the queue for a fair queue policy.
Inbound - Limiter Policies	
<code>traffic-policy limiter &lt;policy-name&gt;</code>	Defines a traffic limiter QoS policy.
<code>traffic-policy limiter &lt;policy-name&gt; description &lt;desc&gt;</code>	Specifies a description for a traffic limiter QoS policy.
Inbound - Limiter Policy Classes	
<code>traffic-policy limiter &lt;policy-name&gt; class &lt;class&gt;</code>	Defines a traffic class for a traffic limiter QoS policy.
<code>traffic-policy limiter &lt;policy-name&gt; class &lt;class&gt; bandwidth</code>	Specifies the bandwidth rate cap for a traffic class.
<code>traffic-policy limiter &lt;policy-name&gt; class &lt;class&gt; burst</code>	Sets the burst size for a traffic class.
<code>traffic-policy limiter &lt;policy-name&gt; class &lt;class&gt; description &lt;desc&gt;</code>	Sets a description for a traffic class.
<code>traffic-policy limiter &lt;policy-name&gt; class &lt;class&gt; match &lt;match-name&gt;</code>	Defines a traffic class matching rule.
<code>traffic-policy limiter &lt;policy-name&gt; class &lt;class&gt; match &lt;match-name&gt; description &lt;desc&gt;</code>	Sets a description for a match rule.

traffic-policy limiter <policy-name> class <class> match <match-name> ether destination <mac-addr>	Specifies a match criterion based on Ethernet destination (MAC) address.
traffic-policy limiter <policy-name> class <class> match <match-name> ether protocol <num>	Specifies a match criterion based on Ethernet packet type.
traffic-policy limiter <policy-name> class <class> match <match-name> ether source <mac-addr>	Specifies a match criterion based on Ethernet source (MAC) address.
traffic-policy limiter <policy-name> class <class> match <match-name> ip destination	Specifies a match criterion based on IP destination information.
traffic-policy limiter <policy-name> class <class> match <match-name> ip dscp <value>	Specifies a match criterion based on the value of the DSCP field.
traffic-policy limiter <policy-name> class <class> match <match-name> ip protocol <proto>	Specifies a match criterion based on the IP protocol.
traffic-policy limiter <policy-name> class <class> match <match-name> ip source	Specifies a match criterion based on source IP information.
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 destination	Specifies a match criterion based on IPv6 destination information.
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 dscp <value>	Specifies a match criterion based on the value of the DSCP field.
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 protocol <proto>	Specifies a match criterion based on the IPv6 protocol.
traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 source	Specifies a match criterion based on source IPv6 information.
traffic-policy limiter <policy-name> class <class> match <match-name> mark <fwmark>	Specifies a match criterion based on the value of the fwmark field.
traffic-policy limiter <policy-name> class <class> match <match-name> vif <vlan-id>	Specifies a match criterion based on VLAN ID.
traffic-policy limiter <policy-name> class <class> priority <priority>	Specifies the order of evaluation of matching rules.
<b>Inbound - Limiter Policy Default Class</b>	
traffic-policy limiter <policy-name> default	Defines a default traffic class for a traffic limiter QoS policy.
traffic-policy limiter <policy-name> default bandwidth	Specifies the bandwidth rate cap for the default traffic class.
traffic-policy limiter <policy-name> default burst	Sets the burst size for the default traffic class.

traffic-policy limiter <policy-name> default priority <priority>	Specifies the order of evaluation of matching rules for the default traffic class.
<b>Outbound - Network Emulator Policies</b>	
traffic-policy network-emulator <policy-name>	Defines a network emulator QoS policy.
traffic-policy network-emulator <policy-name> bandwidth	Specifies the bandwidth limit for all combined traffic constrained by this policy.
traffic-policy network-emulator <policy-name> burst	Sets the burst size for a network emulation QoS policy.
traffic-policy network-emulator <policy-name> description <desc>	Sets a description for a network emulator policy.
traffic-policy network-emulator <policy-name> network-delay	Sets the amount of delay between packets for a network emulation QoS policy.
traffic-policy network-emulator <policy-name> packet-corruption <percent>	Sets the percentage of packets to corrupt in a network emulation QoS policy.
traffic-policy network-emulator <policy-name> packet-loss <percent>	Sets the percentage of packets to drop in a network emulation QoS policy.
traffic-policy network-emulator <policy-name> packet-reordering <percent>	Sets the percentage of packets to reorder in a network emulation QoS policy.
traffic-policy network-emulator <policy-name> queue-limit <limit>	Sets an upper bound for the number of packets allowed in the queue for a network emulation QoS policy.
<b>Outbound - Priority Queue Policies</b>	
traffic-policy priority-queue <policy-name>	Defines a Priority Queue QoS policy.
traffic-policy priority-queue <policy-name> description <desc>	Specifies a description for a priority-queue QoS policy.
<b>Outbound - Priority Queue Policy Classes</b>	
traffic-policy priority-queue <policy-name> class <class>	Defines a traffic class for a round robin QoS policy.
traffic-policy priority-queue <policy-name> class <class> description <desc>	Sets a description for a traffic class.
traffic-policy priority-queue <policy-name> class <class> match <match-name>	Defines a traffic class matching rule.
traffic-policy priority-queue <policy-name> class <class> match <match-name> description <desc>	Sets a description for a match rule.



traffic-policy priority-queue <policy-name> class <class> match <match-name> ether destination <mac-addr>	Specifies a match criterion based on Ethernet destination (MAC) address.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ether protocol <num>	Specifies a match criterion based on Ethernet packet type.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ether source <mac-addr>	Specifies a match criterion based on Ethernet source (MAC) address.
traffic-policy priority-queue <policy-name> class <class> match <match-name> interface <interface>	Specifies a match criterion based on incoming interface.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip destination	Specifies a match criterion based on IP destination information.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip dscp <value>	Specifies a match criterion based on the value of the DSCP field.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip protocol <proto>	Specifies a match criterion based on the IP protocol.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ip source	Specifies a match criterion based on source IP information.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 destination	Specifies a match criterion based on IPv6 destination information.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 dscp <value>	Specifies a match criterion based on the value of the DSCP field.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 protocol <proto>	Specifies a match criterion based on the IPv6 protocol.
traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 source	Specifies a match criterion based on source IPv6 information.
traffic-policy priority-queue <policy-name> class <class> match <match-name> mark <fwmark>	Specifies a match criterion based on the value of the fwmark field.
traffic-policy priority-queue <policy-name> class <class> match <match-name> vif <vlan-id>	Specifies a match criterion based on VLAN ID.
traffic-policy priority-queue <policy-name> class <class> queue-limit <limit>	Specifies the maximum queue size for a traffic class.
traffic-policy priority-queue <policy-name> class <class> queue-type <type>	Specifies the type of queuing to use for a traffic class.
<b>Outbound - Priority Queue Policy Default Class</b>	
traffic-policy priority-queue <policy-name> default	Defines a default priority queue QoS policy.

traffic-policy priority-queue <policy-name> default queue-limit <limit>	Specifies the maximum queue size for the default traffic class.
traffic-policy priority-queue <policy-name> default queue-type <type>	Specifies the type of queuing to use for the default traffic class.
<b>Outbound - Random Detect Policies</b>	
traffic-policy random-detect <policy-name>	Defines a Weighted Random Early Detection (WRED) QoS policy.
traffic-policy random-detect <policy-name> bandwidth	Specifies the bandwidth limit for all combined traffic constrained by this policy.
traffic-policy random-detect <policy-name> description <desc>	Sets a description for a random-detect policy.
traffic-policy random-detect <policy-name> precedence <precedence>	Sets parameters for dropping packets based on precedence for a random-detect policy.
<b>Outbound - Rate Control Policies</b>	
traffic-policy rate-control <policy-name>	Defines a rate controlling QoS policy.
traffic-policy rate-control <policy-name> bandwidth	Specifies the bandwidth limit for all combined traffic constrained by this policy.
traffic-policy rate-control <policy-name> burst	Sets the burst size for a rate controlling QoS policy.
traffic-policy rate-control <policy-name> description <desc>	Sets a description for a rate controlling policy.
traffic-policy rate-control <policy-name> latency	Sets the limit on queue size based on latency for a rate controlling QoS policy.
<b>Outbound - Round Robin Policies</b>	
traffic-policy round-robin <policy-name>	Defines a round robin QoS policy.
traffic-policy round-robin <policy-name> description <desc>	Specifies a description for a round robin QoS policy.
<b>Outbound - Round Robin Policy Classes</b>	
traffic-policy round-robin <policy-name> class <class>	Defines a traffic class for a round robin QoS policy.
traffic-policy round-robin <policy-name> class <class> description <desc>	Sets a description for a traffic class.
traffic-policy round-robin <policy-name> class <class> match <match-name>	Defines a traffic class matching rule.
traffic-policy round-robin <policy-name> class <class> match <match-name> description <desc>	Sets a description for a match rule.

traffic-policy round-robin <policy-name> class <class> match <match-name> ether destination <mac-addr>	Specifies a match criterion based on Ethernet destination (MAC) address.
traffic-policy round-robin <policy-name> class <class> match <match-name> ether protocol <num>	Specifies a match criterion based on Ethernet packet type.
traffic-policy round-robin <policy-name> class <class> match <match-name> ether source <mac-addr>	Specifies a match criterion based on Ethernet source (MAC) address.
traffic-policy round-robin <policy-name> class <class> match <match-name> interface <interface>	Specifies a match criterion based on incoming interface.
traffic-policy round-robin <policy-name> class <class> match <match-name> ip destination	Specifies a match criterion based on IP destination information.
traffic-policy round-robin <policy-name> class <class> match <match-name> ip dscp <value>	Specifies a match criterion based on the value of the DSCP field.
traffic-policy round-robin <policy-name> class <class> match <match-name> ip protocol <proto>	Specifies a match criterion based on the IP protocol.
traffic-policy round-robin <policy-name> class <class> match <match-name> ip source	Specifies a match criterion based on source IP information.
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 destination	Specifies a match criterion based on IPv6 destination information.
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 dscp <value>	Specifies a match criterion based on the value of the DSCP field.
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 protocol <proto>	Specifies a match criterion based on the IPv6 protocol.
traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 source	Specifies a match criterion based on source IPv6 information.
traffic-policy round-robin <policy-name> class <class> match <match-name> mark <fwmark>	Specifies a match criterion based on the value of the fwmark field.
traffic-policy round-robin <policy-name> class <class> match <match-name> vif <vlan-id>	Specifies a match criterion based on VLAN ID.
traffic-policy round-robin <policy-name> class <class> quantum <packets>	Specifies the number of packets that can be sent per scheduling quantum for a traffic class.
traffic-policy round-robin <policy-name> class <class> queue-limit <limit>	Specifies the maximum queue size for a traffic class.
traffic-policy round-robin <policy-name> class <class> queue-type <type>	Specifies the type of queuing to use for a traffic class.

Outbound - Round Robin Policy Default Class	
traffic-policy round-robin <policy-name> default	Defines a default round robin QoS policy.
traffic-policy round-robin <policy-name> default quantum <packets>	Specifies the number of packets that can be sent per scheduling quantum.
traffic-policy round-robin <policy-name> default queue-limit <limit>	Specifies the maximum queue size for the default traffic class.
traffic-policy round-robin <policy-name> default queue-type <type>	Specifies the type of queuing to use for the default traffic class.
Outbound - Shaper Policies	
traffic-policy shaper <policy-name>	Defines a traffic shaping QoS policy.
traffic-policy shaper <policy-name> bandwidth	Specifies the bandwidth available for all combined traffic constrained by this policy.
traffic-policy shaper <policy-name> description <desc>	Specifies a description for a traffic shaper QoS policy.
Outbound - Shaper Policy Classes	
traffic-policy shaper <policy-name> class <class>	Defines a traffic class for a traffic shaper QoS policy.
traffic-policy shaper <policy-name> class <class> bandwidth	Specifies the base guaranteed bandwidth rate for a traffic class.
traffic-policy shaper <policy-name> class <class> burst	Sets the burst size for a traffic class.
traffic-policy shaper <policy-name> class <class> ceiling	Sets a bandwidth ceiling for a traffic class.
traffic-policy shaper <policy-name> class <class> description <desc>	Sets a description for a traffic class.
traffic-policy shaper <policy-name> class <class> match <match-name>	Defines a traffic class matching rule.
traffic-policy shaper <policy-name> class <class> match <match-name> description <desc>	Sets a description for a match rule.
traffic-policy shaper <policy-name> class <class> match <match-name> ether destination <mac-addr>	Specifies a match criterion based on Ethernet destination (MAC) address.
traffic-policy shaper <policy-name> class <class> match <match-name> ether protocol <num>	Specifies a match criterion based on Ethernet packet type.
traffic-policy shaper <policy-name> class <class> match <match-name> ether source <mac-addr>	Specifies a match criterion based on Ethernet source (MAC) address.

traffic-policy shaper <policy-name> class <class> match <match-name> interface <interface>	Specifies a match criterion based on incoming interface.
traffic-policy shaper <policy-name> class <class> match <match-name> ip destination	Specifies a match criterion based on IP destination information.
traffic-policy shaper <policy-name> class <class> match <match-name> ip dscp <value>	Specifies a match criterion based on the value of the DSCP field.
traffic-policy shaper <policy-name> class <class> match <match-name> ip protocol <proto>	Specifies a match criterion based on the IP protocol.
traffic-policy shaper <policy-name> class <class> match <match-name> ip source	Specifies a match criterion based on source IP information.
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 destination	Specifies a match criterion based on IPv6 destination information.
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 dscp <value>	Specifies a match criterion based on the value of the DSCP field.
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 protocol <proto>	Specifies a match criterion based on the IPv6 protocol.
traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 source	Specifies a match criterion based on source IPv6 information.
traffic-policy shaper <policy-name> class <class> match <match-name> mark <fwmark>	Specifies a match criterion based on the value of the fwmark field.
traffic-policy shaper <policy-name> class <class> match <match-name> vif <vlan-id>	Specifies a match criterion based on VLAN ID.
traffic-policy shaper <policy-name> class <class> priority <priority>	Specifies the priority of a traffic class for allocation of extra bandwidth.
traffic-policy shaper <policy-name> class <class> queue-limit <limit>	Specifies the maximum queue size for a traffic class.
traffic-policy shaper <policy-name> class <class> queue-type <type>	Specifies the type of queuing to use for a traffic class.
traffic-policy shaper <policy-name> class <class> set-dscp <value>	Rewrites the DSCP field in packets in this traffic class to the specified value.
<b>Outbound - Shaper Policy Default Class</b>	
traffic-policy shaper <policy-name> default	Defines a default traffic shaper QoS policy.
traffic-policy shaper <policy-name> default bandwidth	Specifies the base guaranteed bandwidth rate for the default traffic class.
traffic-policy shaper <policy-name> default burst	Sets the burst size for the default traffic class.

traffic-policy shaper <policy-name> default ceiling	Sets a bandwidth ceiling for the default traffic class.
traffic-policy shaper <policy-name> default priority <priority>	Specifies the priority of the default traffic class for allocation of extra bandwidth.
traffic-policy shaper <policy-name> default queue-limit <limit>	Specifies the maximum queue size for the default traffic class.
traffic-policy shaper <policy-name> default queue-type <type>	Specifies the type of queuing to use for the default traffic class.
traffic-policy shaper <policy-name> default set-dscp <value>	Rewrites the DSCP field in packets in the default traffic class to the specified value.
<b>Operational Commands</b>	
show incoming	Displays incoming packet actions.
show queueing	Displays outgoing packet actions.

## interfaces <interface> traffic-policy

Applies a QoS policy to the specified interface.

---

### Syntax

```
set interfaces interface traffic-policy [in | out] policy-name
delete interfaces interface traffic-policy [in | out] policy-name
show interfaces interface traffic-policy [in | out] policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
interfaces interface {
    traffic-policy {
        [in | out] policy-name
    }
}
```

---

### Parameters

<i>interface</i>	Mandatory. The type of interface. For detailed keywords and arguments that can be specified as interface types, see the table in the Usage Guidelines below.
<b>in</b>	Apply the QoS policy to inbound traffic on this interface. Note that inbound QoS policies cannot be applied to vif interfaces.
<b>out</b>	Apply the QoS policy to outbound traffic on this interface.
<i>policy-name</i>	The name of the QoS policy to apply to this interface.

---

### Default

None.

---

### Usage Guidelines

Use this command to apply a QoS policy to an interface.

The following table shows the syntax and parameters for supported interface types.

Interface Type	Syntax	Parameters
ADSL Bridged Ethernet	<code>adsl <i>adslx</i> pvc <i>pvc-id</i> bridged-ethernet</code>	<p><i>adslx</i> The name of a Bridged Ethernet- encapsulated DSL interface.</p> <p><i>pvc-id</i> The identifier for the PVC. It can either be the <i>vpi/vci</i> pair or the keyword <b>auto</b>, where <i>vpi</i> is a Virtual Path Index from 0 to 255, <i>vci</i> is a Virtual Circuit Index from from 0 to 65535, and <b>auto</b> directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically.</p>
ADSL Classical IPOA	<code>adsl <i>adslx</i> pvc <i>pvc-id</i> classical-ipoa</code>	<p><i>adslx</i> The name of a Classical IPOA- encapsulated DSL interface.</p> <p><i>pvc-id</i> The identifier for the PVC. It can either be the <i>vpi/vci</i> pair or the keyword <b>auto</b>, where <i>vpi</i> is a Virtual Path Index from 0 to 255, <i>vci</i> is a Virtual Circuit Index from from 0 to 65535, and <b>auto</b> directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically.</p>
ADSL PPPoA	<code>adsl <i>adslx</i> pvc <i>pvc-id</i> pppoa <i>num</i></code>	<p><i>adslx</i> The name of a Classical IPOA- encapsulated DSL interface.</p> <p><i>pvc-id</i> The identifier for the PVC. It can either be the <i>vpi/vci</i> pair or the keyword <b>auto</b>, where <i>vpi</i> is a Virtual Path Index from 0 to 255, <i>vci</i> is a Virtual Circuit Index from from 0 to 65535, and <b>auto</b> directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically.</p> <p><i>num</i> The PPPoA unit number. This number must be unique across all PPPoA interfaces. In addition, only one PPPoA instance can be configured on a PVC. PPPoA units range from 0 to 15 and the resulting interfaces are named pppoa0 to pppoa15.</p>
ADSL PPPoE	<code>adsl <i>adslx</i> pvc <i>pvc-id</i> pppoe <i>num</i></code>	<p><i>adslx</i> The name of a Classical IPOA- encapsulated DSL interface.</p> <p><i>pvc-id</i> The identifier for the PVC. It can either be the <i>vpi/vci</i> pair or the keyword <b>auto</b>, where <i>vpi</i> is a Virtual Path Index from 0 to 255, <i>vci</i> is a Virtual Circuit Index from from 0 to 65535, and <b>auto</b> directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically.</p> <p><i>num</i> The name of a defined PPPoE unit. The range is 0 to 15.</p>
Bonding	<code>bonding <i>bondx</i></code>	<i>bondx</i> The identifier for the bonding interface. Supported values are <b>bond0</b> through <b>bond99</b> .
Bonding Vif	<code>bonding <i>bondx</i> vif <i>vlan-id</i></code>	<p><i>bondx</i> The identifier for the bonding interface. Supported values are <b>bond0</b> through <b>bond99</b>.</p> <p><i>vlan-id</i> The VLAN ID for the vif. The range is 0 to 4094.</p>



Interface Type	Syntax	Parameters
Bridge	bridge <i>brx</i>	<i>brx</i> The name of a Bridge group. The range is <b>br0</b> through <b>br999</b> .
Ethernet	ethernet <i>ethx</i>	<i>ethx</i> The name of an Ethernet interface. The range is <b>eth0</b> through <b>eth23</b> , depending on the physical interfaces available on your system.
Ethernet PPPoE	ethernet <i>ethx</i> pppoe <i>num</i>	<i>ethx</i> The name of an Ethernet interface. The range is <b>eth0</b> through <b>eth23</b> , depending on the physical interfaces available on your system. <i>num</i> The name of a defined PPPoE unit. The range is 0 to 15.
Ethernet Vif	ethernet <i>ethx</i> vif <i>vlan-id</i>	<i>ethx</i> The name of an Ethernet interface. The range is <b>eth0</b> through <b>eth23</b> , depending on the physical interfaces available on your system. <i>vlan-id</i> The VLAN ID for the vif. The range is 0 to 4094.
Ethernet Vif PPPoE	ethernet <i>ethx</i> vif <i>vlan-id</i> pppoe <i>num</i>	<i>ethx</i> The name of an Ethernet interface. The range is <b>eth0</b> through <b>eth23</b> , depending on the physical interfaces available on your system. <i>vlan-id</i> The VLAN ID for the vif. The range is 0 to 4094. <i>num</i> The name of a defined PPPoE unit. The range is 0 to 15.
Loopback	loopback <i>lo</i>	<i>lo</i> The name of the loopback interface.
Multilink	multilink <i>mlx</i> vif <i>1</i>	<i>mlx</i> The identifier of the multilink bundle. You can create up to two multilink bundles. Supported values are <b>ml0</b> ("em ell zero") through <b>ml23</b> ("em ell twenty-three"). <i>1</i> The identifier of the virtual interface. Currently, only one vif is supported for multilink interfaces, and the identifier must be 1. The vif must already have been defined.
OpenVPN	openvpn <i>vtunx</i>	<i>vtunx</i> The identifier for the OpenVPN interface. This may be <b>vtun0</b> to <b>vtunx</b> , where <i>x</i> is a non-negative integer.
Pseudo-Ethernet	pseudo-ethernet <i>pethx</i>	<i>pethx</i> The name of a pseudo-Ethernet interface. The range is <b>peth0</b> through <b>peth999</b> .
Serial Cisco HDLC	serial <i>wanx</i> cisco-hdlc vif <i>1</i>	<i>wanx</i> The serial interface you are configuring: one of <b>wan0</b> through <b>wan23</b> . The interface must already have been defined. <i>1</i> The identifier of the virtual interface. Currently, only one vif is supported for Cisco HDLC interfaces, and the identifier must be 1. The vif must already have been defined.

Interface Type	Syntax	Parameters
Serial Frame Relay	serial <i>wanx</i> frame-relay vif <i>dlci</i>	<p><i>wanx</i> The serial interface you are configuring: one of <b>wan0</b> through <b>wan23</b>. The interface must already have been defined.</p> <p><i>dlci</i> The identifier of the virtual interface. For Frame Relay interfaces, this is the DLCI number for the interface. the range is 16 to 991. The vif must already have been defined.</p>
Serial PPP	serial <i>wanx</i> ppp vif 1	<p><i>wanx</i> The serial interface you are configuring: one of <b>wan0</b> through <b>wan23</b>. The interface must already have been defined.</p> <p>1 The identifier of the virtual interface. Currently, only one vif is supported for point-to-point interfaces, and the identifier must be 1. The vif must already have been defined.</p>
Tunnel	tunnel <i>tunx</i> or tunnel <i>tunx</i> parameters	<i>tunx</i> An identifier for the tunnel interface you are defining. This may be <b>tun0</b> to <b>tunx</b> , where <b>x</b> is a non-negative integer.
Virtual Tunnel	vti <i>vtix</i>	<p><i>vtix</i> An identifier for the virtual tunnel interface you are defining. This may be <b>vti0</b> to <b>vtix</b>, where <b>x</b> is a non-negative integer.</p> <p><b>Note:</b> This interface does not support IPv6.</p>
VRRP	interface <i>parent-if</i> vrrp vrrp-group <i>group</i> interface	<p><i>parent-if</i> The type and identifier of the parent interface; for example, <b>ethernet eth0</b> or <b>bonding bond0</b>.</p> <p><i>group</i> The VRRP group identifier.</p> <p>The name of the VRRP interface is not specified. The system internally constructs the interface name from the parent interface identifier plus the VRRP group number—for example, <b>eth0v99</b>, <b>eth0.15v99</b>, <b>bond0v99</b>, or <b>bond0.15v99</b>. Note that VRRP interfaces support the same feature set as the parent interface does.</p>
Wireless	wireless <i>wlanx</i>	<i>wlanx</i> The identifier for the wireless interface you are using. This may be <b>wlan0</b> to <b>wlan999</b> .
Wireless Modem	wirelessmodem <i>wlmx</i>	<i>wlmx</i> The identifier for the wireless modem interface you are using. This may be <b>wlm0</b> to <b>wlm999</b> .

Use the **set** form of this command to apply the QoS policy to the interface.

Use the **delete** form of this command to remove the QoS policy from the interface.

Use the **show** form of this command to display QoS policy configuration for an interface.

## show incoming

Displays incoming packet actions.

---

### Syntax

```
show incoming [interface-type [interface]]
```

---

### Command Mode

Operational mode.

---

### Parameters

<i>interface-type</i>	Optional. The type of interface whose incoming policies you wish to see. Possible values include <b>adsl</b> , <b>bonding</b> , <b>ethernet</b> , <b>input</b> , <b>pppoe</b> , <b>pseudo-ethernet</b> , <b>serial</b> , <b>tunnel</b> , <b>vti</b> , and <b>wireless</b> .
<i>interface</i>	Optional. The specific interface (e.g. eth0).

---

### Default

None.

---

### Usage Guidelines

Use this command to display incoming packet actions.

---

### Examples

[Example 3-1](#) shows all incoming packet actions.

Example 3-1 “show incoming”: Displaying all incoming packet actions.

```
vyatta@vyatta:~$ show incoming
Interface Action    Received  Dropped  Overlimit
eth0      limiter    32        10        0
eth2      redirect   64        0         0
vyatta@vyatta:~$
```

## show queueing

Displays outgoing packet actions.

---

### Syntax

```
show queueing [interface-type [interface]]
```

---

### Command Mode

Operational mode.

---

### Parameters

---

<i>interface-type</i>	Optional. The type of interface whose QoS policies you wish to see. Possible values include <b>adsl</b> , <b>bonding</b> , <b>ethernet</b> , <b>input</b> , <b>pppoe</b> , <b>pseudo-ethernet</b> , <b>serial</b> , <b>tunnel</b> , <b>vti</b> , and <b>wireless</b> .
<i>interface</i>	Optional. The specific interface (e.g. eth0).

---

---

### Default

None.

---

### Usage Guidelines

Use this command to display outgoing packet actions.

---

### Examples

[Example 3-2](#) shows all outgoing QoS policies.

Example 3-2 “show queueing”: Displaying all outgoing QoS policies.

---

```
vyatta@vyatta:~$ show queueing
Interface Policy          Sent    Dropped  Overlimit
eth0      weighted-random        0         0         0
eth1      default             36888        0         0
eth2      default              408         0         0
ifb0      shaper                92         0         0
vyatta@vyatta:~$
```

---

Example 3-3 shows specific QoS policies.

Example 3-3 “show queueing ethernet eth0”: Displaying QoS policies on a specific interface.

---

```
vyatta@vyatta:~$ show queueing ethernet eth0
eth0 Output queue:
Class      Policy      Sent      Dropped    Overlimit
   1       shaper      106384      0           0
  8001    fair-queue      48286      0           0
  8002    fair-queue      58098      0           0
  8003    drop-tail         0          0           0
vyatta@vyatta:~$
```

---

## traffic-policy drop-tail <policy-name>

Defines a drop tail (pure FIFO) QoS policy.

---

### Syntax

```
set traffic-policy drop-tail policy-name
delete traffic-policy drop-tail policy-name
show traffic-policy drop-tail policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    drop-tail policy-name {
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the drop tail policy.
--------------------	--

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a drop tail QoS policy. The drop tail policy acts on outbound traffic only. The policy name must be unique and not used with other QoS policy commands.

The drop tail policy provides a pure First In First Out (FIFO) queueing mechanism.

Use the **set** form of this command to create a drop tail policy.

Use the **delete** form of this command to remove a drop tail policy.

Use the **show** form of this command to display drop tail policy configuration.

## traffic-policy drop-tail <policy-name> description <desc>

Sets a description for a drop tail policy.

---

### Syntax

```
set traffic-policy drop-tail policy-name description desc  
delete traffic-policy drop-tail policy-name description  
show traffic-policy drop-tail policy-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    drop-tail policy-name {  
        description desc  
    }  
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the drop tail policy.
<i>desc</i>	Mandatory. The description for this drop tail policy.

---

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a drop tail policy.

Use the **set** form of this command to specify a description.

Use the **delete** form of this command to remove a description.

Use the **show** form of this command to display description configuration.

## traffic-policy drop-tail <policy-name> queue-limit <limit>

Sets an upper bound for the number of packets allowed in the queue for a drop tail policy.

---

### Syntax

```
set traffic-policy drop-tail policy-name queue-limit limit
delete traffic-policy drop-tail policy-name queue-limit
show traffic-policy drop-tail policy-name queue-limit
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    drop-tail policy-name {
        queue-limit limit
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the drop tail policy.
<i>limit</i>	Optional. The maximum queue size, in packets. The range is 0 to 4294967295. The default is the same as the underlying hardware transmit queue length. For Ethernet this is typically 1000 packets.

---

### Default

For Ethernet the queue limit is typically 1000 packets.

---

### Usage Guidelines

Use this command to set the maximum number of packets that can wait in a queue for this queuing policy. If maximum queue size is reached, the system begins dropping packets.

Use the **set** form of this command to set the queue limit.



Use the **delete** form of this command to restore the default queue limit.

Use the **show** form of this command to display queue limit configuration.

## traffic-policy fair-queue <policy-name>

Defines a fair queue QoS policy.

---

### Syntax

```
set traffic-policy fair-queue policy-name
delete traffic-policy fair-queue policy-name
show traffic-policy fair-queue policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    fair-queue policy-name {
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the fair queue policy.
--------------------	---

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a fair queue (FQ) QoS policy. The FQ policy acts on outbound traffic only. The policy name must be unique and not used with other QoS policy commands.

The Vyatta system uses Stochastic Fair Queuing, which is one of a number of FQ algorithms aiming to provide per-flow-based fairness. The FQ algorithm attempts to provide fair access to network resources and prevent any one flow from consuming an inordinate amount of output port bandwidth.

In Stochastic Fair Queuing, bandwidth is divided into separate hash buckets based on the combination of IP protocol, source, and destination address such that no single flow receives an unfair portion of bandwidth.

Use the **set** form of this command to create an FQ policy.

Use the **delete** form of this command to remove an FQ policy.

Use the **show** form of this command to display FQ policy configuration.

## traffic-policy fair-queue <policy-name> description <desc>

Sets a description for a fair queue policy.

---

### Syntax

```
set traffic-policy fair-queue policy-name description desc
delete traffic-policy fair-queue policy-name description
show traffic-policy fair-queue policy-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    fair-queue policy-name {
        description desc
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the fair queue policy.
<i>desc</i>	Mandatory. The description for this fair queue policy.

---

### Default

None.

---

### Usage Guidelines

- Use this command to record a description for a fair queue policy.
- Use the **set** form of this command to specify a description.
- Use the **delete** form of this command to remove a description.
- Use the **show** form of this command to display description configuration.

## traffic-policy fair-queue <policy-name> hash-interval <seconds>

Specifies the interval between flow hash function updates for a fair queue policy.

---

### Syntax

```
set traffic-policy fair-queue policy-name hash-interval seconds
delete traffic-policy fair-queue policy-name hash-interval
show traffic-policy fair-queue policy-name hash-interval
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    fair-queue policy-name {
        hash-interval seconds
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the fair queue policy.
<i>seconds</i>	Mandatory. The rehash interval, in seconds. The range is 0 to 4294967295, where 0 means the hash function is never updated.

---

### Default

The hash function is never updated.

---

### Usage Guidelines

Use this command to set the interval at which the flow hash function is updated.

Updating the hash function at intervals increases security and prevents attacks based on an attacker determining the hash bucket for traffic flows and sending spoofed packets based on that information.

Use the set form of this command to specify a flow hash update interval.

Use the **delete** form of this command to restore the default hash interval.

Use the **show** form of this command to display hash interval configuration.

## traffic-policy fair-queue <policy-name> queue-limit <limit>

Sets an upper bound for the number of packets allowed in the queue for a fair queue policy.

---

### Syntax

```
set traffic-policy fair-queue policy-name queue-limit limit
delete traffic-policy fair-queue policy-name queue-limit
show traffic-policy fair-queue policy-name queue-limit
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    fair-queue policy-name {
        queue-limit limit
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the fair queue policy.
<i>limit</i>	Mandatory. The maximum queue size, in packets. The range is 0 to 4294967295. The default is 127.

---

---

### Default

A queue is not permitted to exceed 127 packets.

---

### Usage Guidelines

Use this command to set the maximum number of packets that can wait in a queue for this queuing policy. If maximum queue size is reached, the system begins dropping packets.

Use the **set** form of this command to set the queue limit.

Use the **delete** form of this command to restore the default queue limit.

Use the **show** form of this command to display queue limit configuration.



## traffic-policy limiter <policy-name>

Defines a traffic limiter QoS policy.

---

### Syntax

```
set traffic-policy limiter policy-name
delete traffic-policy limiter policy-name
show traffic-policy limiter policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
--------------------	---

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a traffic limiter QoS policy. Traffic limiter policy acts on inbound traffic only. The policy name must be unique and not used with other QoS policy commands.

Traffic is evaluated against the matching rules which are similar to outbound traffic shaper. Any traffic that matches no rules is let through unrestricted. Any traffic that exceeds the bandwidth limits is dropped.

Use the **set** form of this command to create a traffic limiter QoS policy.

Use the **delete** form of this command to remove a traffic limiter QoS policy.

Use the **show** form of this command to display traffic limiter QoS policy configuration.

## traffic-policy limiter <policy-name> class <class>

Defines a traffic class for a traffic limiter QoS policy.

---

### Syntax

```
set traffic-policy limiter policy-name class class
delete traffic-policy limiter policy-name class class
show traffic-policy limiter policy-name class class
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
        class class {
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.

---

### Default

None.

---

### Usage Guidelines

Use this command to define a traffic class for a traffic limiter QoS policy. This allows packets to be grouped into various traffic classes, which can be treated with different levels of service.

Use the set form of this command to create a traffic class in a traffic limiter QoS policy.

Use the **delete** form of this command to remove a traffic class from a traffic limiter QoS policy.

Use the **show** form of this command to display traffic class configuration within a traffic limiter QoS policy.

## traffic-policy limiter <policy-name> class <class> bandwidth

Specifies the bandwidth rate cap for a traffic class.

---

### Syntax

```
set traffic-policy limiter policy-name class class bandwidth [rate | rate-suffix]  
delete traffic-policy limiter policy-name class class bandwidth  
show traffic-policy limiter policy-name class class bandwidth
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        class class {  
            bandwidth [rate | rate-suffix]  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>rate</i>	The bandwidth, specified in kilobits per second.

---

<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported: <b>kbit</b> : Kilobits per second. <b>mbit</b> : Megabits per second. <b>gbit</b> : Gigabits per second. <b>kbps</b> : Kilobytes per second. <b>mbps</b> : Megabytes per second. <b>gbps</b> : Gigabytes per second.
--------------------	---

---

---

### Default

None. This value must be set.

---

### Usage Guidelines

Use this command to set a bandwidth cap for a traffic class.

Use the **set** form of this command to set the available bandwidth for the traffic class.

Use the **delete** form of this command to restore the default available bandwidth for the traffic class.

Use the **show** form of this command to display class bandwidth configuration.

## traffic-policy limiter <policy-name> class <class> burst

Sets the burst size for a traffic class.

---

### Syntax

```
set traffic-policy limiter policy-name class class burst [num | num-suffix]  
delete traffic-policy limiter policy-name class class burst  
show traffic-policy limiter policy-name class class burst
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        class class {  
            burst [num | num-suffix]  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>num</i>	The burst size, specified in bytes.
<i>num-suffix</i>	The burst size, specified as a number and a scaling suffix (for example, 10mb). The following suffixes are supported: <b>kb</b> : Kilobytes. <b>mb</b> : Megabytes. <b>gb</b> : Gigabytes.

---

### Default

The burst size is 15 kilobytes.

---

### Usage Guidelines

Use this command to set the burst size for the traffic class. This is the maximum amount of traffic that may be sent at a given time.

Use the **set** form of this command to specify the burst size for a traffic class.

Use the **delete** form of this command to restore the default burst size for a traffic class.

Use the **show** form of this command to display traffic class burst size configuration.



## traffic-policy limiter <policy-name> class <class> description <desc>

Sets a description for a traffic class.

---

### Syntax

```
set traffic-policy limiter policy-name class class description desc  
delete traffic-policy limiter policy-name class class description  
show traffic-policy limiter policy-name class class description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        class class {  
            description desc  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>desc</i>	The description for this traffic class.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic class.

Use the **set** form of this command to specify a description.

Use the **delete** form of this command to remove a description.

Use the **show** form of this command to display description configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name>

Defines a traffic class matching rule.

---

### Syntax

```
set traffic-policy limiter policy-name class class match match-name
delete traffic-policy limiter policy-name class class match match-name
show traffic-policy limiter policy-name class class match match-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
        class class {
            match match-name {
            }
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.

---

### Default

None.

---

### Usage Guidelines

Use this command to define a rule setting out the match conditions for membership in a traffic class.

Use the **set** form of this command to create the traffic class matching rule. Note that you cannot use **set** to change the name of an existing traffic class matching rule. To change the rule, delete it and re-create it.

Use the **delete** form of this command to remove the traffic class matching rule configuration node.

Use the **show** form of this command to display traffic class matching rule configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> description <desc>

Sets a description for a match rule.

---

### Syntax

```
set traffic-policy limiter policy-name class class match match-name description desc  
delete traffic-policy limiter policy-name class class match match-name description  
show traffic-policy limiter policy-name class class match match-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        class class {  
            match match-name {  
                description desc  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>desc</i>	The description for this match.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic class matching rule.

Use the **set** form of this command to set the description.

Use the **delete** form of this command to remove the description.

Use the **show** form of this command to display description configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> ether destination <mac-addr>

Specifies a match criterion based on Ethernet destination (MAC) address.

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ether destination mac-addr
```

```
delete traffic-policy limiter policy-name class class match match-name ether destination
```

```
show traffic-policy limiter policy-name class class match match-name ether destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  limiter policy-name {
    class class {
      match match-name {
        ether {
          destination mac-addr
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>mac-addr</i>	Performs a match based on the destination MAC address on the interface the policy is applied to. The format is 6 colon-separated 8-bit numbers in hexadecimal; for example, 00:0a:59:9a:f2:ba.

---

### Default

If not set, packets are not matched against the destination MAC address.

---

### Usage Guidelines

Use this command to define a match condition based on destination MAC address for a traffic class.

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.



## traffic-policy limiter <policy-name> class <class> match <match-name> ether protocol <num>

Specifies a match criterion based on Ethernet packet type.

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ether protocol
num
```

```
delete traffic-policy limiter policy-name class class match match-name ether protocol
```

```
show traffic-policy limiter policy-name class class match match-name ether protocol
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
        class class {
            match match-name {
                ether {
                    protocol num
                }
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>num</i>	Performs a match based on the Ethernet packet type (i.e. protocol number). The range is 0 to 65535.

---

### Default

If not set, packets are not matched against the Ethernet packet type.

---

### Usage Guidelines

Use this command to define a match condition based on Ethernet packet type for a traffic class.

Use the **set** form of this command to specify the packet type to be matched.

Use the **delete** form of this command to remove packet type as a match condition.

Use the **show** form of this command to display packet type match condition configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> ether source <mac-addr>

Specifies a match criterion based on Ethernet source (MAC) address.

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ether source mac-addr
```

```
delete traffic-policy limiter policy-name class class match match-name ether source
```

```
show traffic-policy limiter policy-name class class match match-name ether source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  limiter policy-name {
    class class {
      match match-name {
        ether {
          source mac-addr
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>mac-addr</i>	Performs a match based on the source MAC address. The format is 6 colon-separated 8-bit numbers in hexadecimal; for example, 00:0a:59:9a:f2:ba.

---

### Default

If not set, packets are not matched against the source MAC address.

---

### Usage Guidelines

Use this command to define a match condition based on source MAC address for a traffic class.

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> ip destination

Specifies a match criterion based on IP destination information.

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ip destination
{address ipv4net | port port}
```

```
delete traffic-policy limiter policy-name class class match match-name ip destination
[address | port]
```

```
show traffic-policy limiter policy-name class class match match-name ip destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  limiter policy-name {
    class class {
      match match-name {
        ip {
          destination {
            address ipv4net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>ipv4net</i>	Performs a match based on the destination IP subnet address.

---

<i>port</i>	Performs a match based on destination port. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.
-------------	---

---

---

### Default

If not set, packets are not matched against destination information.

---

### Usage Guidelines

Use this command to define a match condition based on destination subnet address and/or port for a traffic class.

You can match packets based on a destination represented by either or both of IP subnet address and destination port(s).

Note that you are not able to match on both “ip” and “vif” (or “interface”) inside the same traffic limiter configuration.

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> ip dscp <value>

Specifies a match criterion based on the value of the DSCP field.

---

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ip dscp value
delete traffic-policy limiter policy-name class class match match-name ip dscp
show traffic-policy limiter policy-name class class match match-name ip dscp
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
        class class {
            match match-name {
                ip {
                    dscp value
                }
            }
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>value</i>	Performs a match based on the specified value. This value is compared with the value in the DSCP field of the ToS byte in the IP header. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <i>/etc/iproute2/rt_dsfield</i> (for example, <i>lowdelay</i> ).

---

---

### Default

If not set, packets are not matched against DSCP value.

---

### Usage Guidelines

Use this command to define a match condition based on the Differentiated Services Code Point (DSCP) field.

The DSCP field is a 6-bit field in the Type of Service (ToS) byte of the IP header. It provides a way of marking packets in order to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

Note that you are not able to match on both “ip” and “vif” (or “interface”) inside the same traffic limiter configuration.

Use the **set** form of this command to set a match condition based on DSCP value.

Use the **delete** form of this command to remove DSCP as a match condition.

Use the **show** form of this command to display DSCP value configuration.



## traffic-policy limiter <policy-name> class <class> match <match-name> ip protocol <proto>

Specifies a match criterion based on the IP protocol.

---

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ip protocol proto  
delete traffic-policy limiter policy-name class class match match-name ip protocol  
show traffic-policy limiter policy-name class class match match-name ip protocol
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        class class {  
            match match-name {  
                ip {  
                    protocol proto  
                }  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>proto</i>	Performs a match based on the protocol name (for example, <b>icmp</b> ) or number, as assigned by the IANA.

---

### Default

If not set, packets are not matched against IP protocol.

---

### Usage Guidelines

Use this command to define a match condition for a traffic class based on protocol.

Note that you are not able to match on both “ip” and “vif” (or “interface”) inside the same traffic limiter configuration.

Use the **set** form of this command to set a match condition based on protocol.

Use the **delete** form of this command to remove protocol value as a match condition.

Use the **show** form of this command to match condition protocol configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> ip source

Specifies a match criterion based on source IP information.

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ip source
{address ipv4net | port port}

delete traffic-policy limiter policy-name class class match match-name ip source
{address | port}

show traffic-policy limiter policy-name class class match match-name ip source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  limiter policy-name {
    class class {
      match match-name {
        ip {
          source {
            address ipv4net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>ipv4net</i>	The source IP subnet address to match for this rule.

---

<i>port</i>	The source port to match for this rule. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.
-------------	---

---

---

### Default

If not set, packets are not matched against IP source information.

---

### Usage Guidelines

Use this command to define a match condition based on source subnet address and/or port for a traffic class.

You can match packets based on a source represented by either or both of IP subnet address and destination port(s).

Note that you are not able to match on both “ip” and “vif” (or “interface”) inside the same traffic limiter configuration.

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 destination

Specifies a match criterion based on IPv6 destination information.

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ipv6 destination
{address ipv6net | port port}
```

```
delete traffic-policy limiter policy-name class class match match-name ipv6
destination [address | port]
```

```
show traffic-policy limiter policy-name class class match match-name ipv6
destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  limiter policy-name {
    class class {
      match match-name {
        ipv6 {
          destination {
            address ipv6net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.

---

<i>ipv6net</i>	Performs a match based on the destination IPv6 subnet address.
<i>port</i>	Performs a match based on destination port. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.

---

---

### Default

If not set, packets are not matched against destination information.

---

### Usage Guidelines

Use this command to define a match condition based on destination subnet address and/or port for a traffic class.

You can match packets based on a destination represented by either or both of IPv6 subnet address and destination port(s).

Note that you are not able to match on both “ipv6” and “vif” (or “interface”) inside the same traffic limiter configuration.

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 dscp <value>

Specifies a match criterion based on the value of the DSCP field.

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ipv6 dscp value
delete traffic-policy limiter policy-name class class match match-name ipv6 dscp
show traffic-policy limiter policy-name class class match match-name ipv6 dscp
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
        class class {
            match match-name {
                ipv6 {
                    dscp value
                }
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>value</i>	Performs a match based on the specified value. This value is compared with the value in the DSCP field of the ToS byte in the IP header. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <i>/etc/iproute2/rt_dsfield</i> (for example, <i>lowdelay</i> ).

---

### Default

If not set, packets are not matched against DSCP value.

---

### Usage Guidelines

Use this command to define a match condition based on the Differentiated Services Code Point (DSCP) field.

The DSCP field is a 6-bit field in the Type of Service (ToS) byte of the IP header. It provides a way of marking packets in order to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

Note that you are not able to match on both “ipv6” and “vif” (or “interface”) inside the same traffic limiter configuration.

Use the **set** form of this command to set a match condition based on DSCP value.

Use the **delete** form of this command to remove DSCP as a match condition.

Use the **show** form of this command to display DSCP value configuration.



## traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 protocol <proto>

Specifies a match criterion based on the IPv6 protocol.

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ipv6 protocol
proto
```

```
delete traffic-policy limiter policy-name class class match match-name ipv6 protocol
```

```
show traffic-policy limiter policy-name class class match match-name ipv6 protocol
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
        class class {
            match match-name {
                ipv6 {
                    protocol proto
                }
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>proto</i>	Performs a match based on the protocol name (for example, <b>icmp</b> ) or number, as assigned by the IANA.

---

### Default

If not set, packets are not matched against IP protocol.

---

### Usage Guidelines

Use this command to define a match condition for a traffic class based on protocol.

Note that you are not able to match on both “ipv6” and “vif” (or “interface”) inside the same traffic limiter configuration.

Use the **set** form of this command to set a match condition based on protocol.

Use the **delete** form of this command to remove protocol value as a match condition.

Use the **show** form of this command to match condition protocol configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> ipv6 source

Specifies a match criterion based on source IPv6 information.

### Syntax

```
set traffic-policy limiter policy-name class class match match-name ipv6 source
{address ipv6net | port port}

delete traffic-policy limiter policy-name class class match match-name ipv6 source
{address | port}

show traffic-policy limiter policy-name class class match match-name ipv6 source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  limiter policy-name {
    class class {
      match match-name {
        ipv6 {
          source {
            address ipv6net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>ipv6net</i>	The source IPv6 subnet address to match for this rule.

---

<i>port</i>	The source port to match for this rule. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.
-------------	---

---

---

### Default

If not set, packets are not matched against IP source information.

---

### Usage Guidelines

Use this command to define a match condition based on source subnet address and/or port for a traffic class.

You can match packets based on a source represented by either or both of IPv6 subnet address and destination port(s).

Note that you are not able to match on both “ipv6” and “vif” (or “interface”) inside the same traffic limiter configuration.

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> mark <fwmark>

Specifies a match criterion based on the value of the fwmark field.

---

### Syntax

```
set traffic-policy limiter policy-name class class match match-name mark fwmark  
delete traffic-policy limiter policy-name class class match match-name mark  
show traffic-policy limiter policy-name class class match match-name mark
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        class class {  
            match match-name {  
                mark fwmark  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>fwmark</i>	Performs a match based on the fwmark field. The range is 0 to 4294967295.

---

### Default

If not set, packets are not matched against fwmark.

---

### Usage Guidelines

Use this command to define a match condition based on the fwmark field for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a fwmark value to be matched.

Use the **delete** form of this command to remove mark as a match condition.

Use the **show** form of this command to display the mark match condition configuration.

## traffic-policy limiter <policy-name> class <class> match <match-name> vif <vlan-id>

Specifies a match criterion based on VLAN ID.

---

### Syntax

```
set traffic-policy limiter policy-name class class match match-name vif vlan-id  
delete traffic-policy limiter policy-name class class match match-name vif  
show traffic-policy limiter policy-name class class match match-name vif
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        class class {  
            match match-name {  
                vif vlan-id  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>vlan-id</i>	Performs a match based on VLAN ID. The range is 1 to 4096.

---

### Default

If not set, packets are not matched against VLAN ID.

---

### Usage Guidelines

Use this command to define a match condition based on VLAN ID for a traffic class.

Note that you are not able to match on both “ip” and “vif” (or “interface”) inside the same traffic limiter configuration.

Use the **set** form of this command to specify a VLAN ID to be matched.

Use the **delete** form of this command to remove VLAN ID as a match condition.

Use the **show** form of this command to display VLAN ID match condition configuration.



## traffic-policy limiter <policy-name> class <class> priority <priority>

Specifies the order of evaluation of matching rules.

---

### Syntax

```
set traffic-policy limiter policy-name class class priority priority
delete traffic-policy limiter policy-name class class priority
show traffic-policy limiter policy-name class class priority
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
        class class {
            priority priority
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 4095.
<i>priority</i>	The priority with which matching rules are evaluated. The range is 0 to 20, where the higher the number the lower the priority. The default is 20.

---

### Default

Traffic classes are assigned a priority of 20.

---

### Usage Guidelines

Use this command to set the priority with which matching rules are evaluated.

Use the **set** form of this command to specify priority for a traffic class.

Use the **delete** form of this command to restore the default priority for a traffic class.

Use the **show** form of this command to display traffic class priority configuration.

## traffic-policy limiter <policy-name> default

Defines a default traffic class for a traffic limiter QoS policy.

---

### Syntax

```
set traffic-policy limiter policy-name default
delete traffic-policy limiter policy-name default
show traffic-policy limiter policy-name default
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
        default {
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
--------------------	---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a default traffic class for a traffic limiter QoS policy. All traffic that doesn't match any of the other classes defined for this policy are handled by the default class.

Use the **set** form of this command to create a default traffic class in a traffic limiter QoS policy.

Use the **delete** form of this command to remove the default traffic class from a traffic limiter QoS policy.

Use the **show** form of this command to display the default traffic class configuration within a traffic limiter QoS policy.

## traffic-policy limiter <policy-name> default bandwidth

Specifies the bandwidth rate cap for the default traffic class.

---

### Syntax

```
set traffic-policy limiter policy-name default bandwidth [rate | rate-suffix]  
delete traffic-policy limiter policy-name default bandwidth  
show traffic-policy limiter policy-name default bandwidth
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        default {  
            bandwidth [rate | rate-suffix]  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>rate</i>	The bandwidth, specified in kilobits per second.
<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported: <b>kbit</b> : Kilobits per second. <b>mbit</b> : Megabits per second. <b>gbit</b> : Gigabits per second. <b>kbps</b> : Kilobytes per second. <b>mbps</b> : Megabytes per second. <b>gbps</b> : Gigabytes per second.

---

---

### Default

None. This value must be set.

---

### Usage Guidelines

Use this command to set a bandwidth cap for the default traffic class.

Use the **set** form of this command to set the available bandwidth for the default traffic class.

Use the **delete** form of this command to restore the default available bandwidth for the default traffic class.

Use the **show** form of this command to display default class bandwidth configuration.

## traffic-policy limiter <policy-name> default burst

Sets the burst size for the default traffic class.

---

### Syntax

```
set traffic-policy limiter policy-name default burst [num | num-suffix]
```

```
delete traffic-policy limiter policy-name default burst
```

```
show traffic-policy limiter policy-name default burst
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        default {  
            burst [num | num-suffix]  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>num</i>	The burst size, specified in bytes.
<i>num-suffix</i>	The burst size, specified as a number and a scaling suffix (for example, 10mb). The following suffixes are supported: <b>kb</b> : Kilobytes. <b>mb</b> : Megabytes. <b>gb</b> : Gigabytes.

---

### Default

The burst size is 15 kilobytes.

---

### Usage Guidelines

Use this command to set the burst size for the default traffic class. This is the maximum amount of traffic that may be sent at a given time.

Use the **set** form of this command to specify the burst size for the default traffic class.

Use the **delete** form of this command to restore the default burst size for the default traffic class.

Use the **show** form of this command to display default traffic class burst size configuration.



## traffic-policy limiter <policy-name> default priority <priority>

Specifies the order of evaluation of matching rules for the default traffic class.

---

### Syntax

```
set traffic-policy limiter policy-name default priority priority
delete traffic-policy limiter policy-name default priority
show traffic-policy limiter policy-name default priority
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    limiter policy-name {
        default {
            priority priority
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>priority</i>	The priority with which matching rules are evaluated. The range is 0 to 20, where the higher the number the lower the priority. The default is 20.

---

### Default

Traffic classes are assigned a priority of 20.

---

### Usage Guidelines

Use this command to set the priority with which matching rules are evaluated.  
Use the **set** form of this command to specify priority for the default traffic class.

Use the **delete** form of this command to restore the default priority for the default traffic class.

Use the **show** form of this command to display default traffic class priority configuration.

## traffic-policy limiter <policy-name> description <desc>

Specifies a description for a traffic limiter QoS policy.

---

### Syntax

```
set traffic-policy limiter policy-name description desc  
delete traffic-policy limiter policy-name description  
show traffic-policy limiter policy-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    limiter policy-name {  
        description desc  
    }  
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the traffic limiting policy.
<i>desc</i>	The description for this traffic limiter policy.

---

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic limiter policy.

Use the **set** form of this command to specify a description for a traffic limiter policy.

Use the **delete** form of this command to remove a description from a traffic limiter policy.

Use the **show** form of this command to display description configuration for a traffic limiter policy.

## traffic-policy network-emulator <policy-name>

Defines a network emulator QoS policy.

---

### Syntax

```
set traffic-policy network-emulator policy-name
delete traffic-policy network-emulator policy-name
show traffic-policy network-emulator policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    network-emulator policy-name {
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the network emulator policy.
--------------------	---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a network emulation QoS policy used to emulate WAN networks. The policy name must be unique and not used with other QoS policy commands.

Use the **set** form of this command to create a network emulator QoS policy.

Use the **delete** form of this command to remove a network emulator QoS policy.

Use the **show** form of this command to display network emulator QoS policy configuration.

## traffic-policy network-emulator <policy-name> bandwidth

Specifies the bandwidth limit for all combined traffic constrained by this policy.

---

### Syntax

```
set traffic-policy network-emulator policy-name bandwidth [rate | rate-suffix]  
delete traffic-policy network-emulator policy-name bandwidth  
show traffic-policy network-emulator policy-name bandwidth
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    network-emulator policy-name {  
        bandwidth [rate | rate-suffix]  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the network emulator policy.
<i>rate</i>	Optional. The bandwidth, specified in kilobits per second.
<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported:  kbit: Kilobits per second. mbit: Megabits per second. gbit: Gigabits per second. kbits: Kilobytes per second. mbps: Megabytes per second. gbps: Gigabytes per second.

---

---

### Default

Traffic flows at maximum speed.

---

### Usage Guidelines

Use this command to set bandwidth constraints for a network emulator QoS policy. This is the maximum bandwidth available to the network emulator policy.

Use the **set** form of this command to specify bandwidth constraints for the policy.

Use the **delete** form of this command to restore default bandwidth constraints for the policy.

Use the **show** form of this command to display policy bandwidth configuration.

## traffic-policy network-emulator <policy-name> burst

Sets the burst size for a network emulation QoS policy.

---

### Syntax

```
set traffic-policy network-emulator policy-name burst [num | num-suffix]  
delete traffic-policy network-emulator policy-name burst  
show traffic-policy network-emulator policy-name burst
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    network-emulator policy-name {  
        burst [num | num-suffix]  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the network emulator policy.
<i>num</i>	The burst size, specified in bytes.
<i>num-suffix</i>	The burst size, specified as a number and a scaling suffix (for example, 10mb). The following suffixes are supported:  kb: Kilobytes. mb: Megabytes. gb: Gigabytes.

---

### Default

The default burst size is 15 kilobytes.

---

### Usage Guidelines

Use this command to set the burst size for a network emulator QoS policy. This is the maximum amount of traffic that may be sent at a given time and is only used with the **bandwidth** parameter.

Use the **set** form of this command to specify the burst size for a network emulator QoS policy.

Use the **delete** form of this command to restore the default burst size for a network emulator QoS policy.

Use the **show** form of this command to display network emulator burst size configuration.



## traffic-policy network-emulator <policy-name> description <desc>

Sets a description for a network emulator policy.

---

### Syntax

```
set traffic-policy network-emulator policy-name description desc  
delete traffic-policy network-emulator policy-name description  
show traffic-policy network-emulator policy-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    network-emulator policy-name {  
        description desc  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the network emulator policy.
<i>desc</i>	Mandatory. The description for this network emulator policy.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a network emulator policy.

Use the **set** form of this command to specify a description.

Use the **delete** form of this command to remove a description.

Use the **show** form of this command to display description configuration.

## traffic-policy network-emulator <policy-name> network-delay

Sets the amount of delay between packets for a network emulation QoS policy.

---

### Syntax

```
set traffic-policy network-emulator policy-name network-delay [num | num-suffix]  
delete traffic-policy network-emulator policy-name network-delay  
show traffic-policy network-emulator policy-name network-delay
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    network-emulator policy-name {  
        network-delay [num | num-suffix]  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the network emulator policy.
<i>num</i>	The latency, specified in milliseconds.
<i>num-suffix</i>	The latency, specified as a time and a scaling suffix (for example, 10ms). The following suffixes are supported:  secs: Seconds. ms: Milliseconds. us: Microseconds.

---

### Default

None.

---

### Usage Guidelines

Use this command to set the network delay for a network emulator QoS policy. This is the amount of delay that will be added between packets.

Use the **set** form of this command to specify the network delay for a network emulator QoS policy.

Use the **delete** form of this command to restore the default network delay for a network emulator QoS policy.

Use the **show** form of this command to display network delay configuration.

## traffic-policy network-emulator <policy-name> packet-corruption <percent>

Sets the percentage of packets to corrupt in a network emulation QoS policy.

---

### Syntax

```
set traffic-policy network-emulator policy-name packet-corruption percent[%]  
delete traffic-policy network-emulator policy-name packet-corruption  
show traffic-policy network-emulator policy-name packet-corruption
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    network-emulator policy-name {  
        packet-corruption percent  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the network emulator policy.
<i>percent</i>	The percentage of packets to corrupt on a random basis.

---

### Default

No packets will be corrupted (i.e. 0%).

---

### Usage Guidelines

Use this command to set the percentage of packets to corrupt in a network emulator QoS policy. This emulates link problems which cause packet corruption by flipping one random bit in the packet and not modifying the checksum.

Use the set form of this command to specify the percentage of packets to randomly corrupt for a network emulator QoS policy.

Use the **delete** form of this command to restore the default percentage of packets to corrupt for a network emulator QoS policy.

Use the **show** form of this command to display packet corruption configuration.

## traffic-policy network-emulator <policy-name> packet-loss <percent>

Sets the percentage of packets to drop in a network emulation QoS policy.

---

### Syntax

```
set traffic-policy network-emulator policy-name packet-loss percent[%]  
delete traffic-policy network-emulator policy-name packet-loss  
show traffic-policy network-emulator policy-name packet-loss
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    network-emulator policy-name {  
        packet-loss percent  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the network emulator policy.
<i>percent</i>	The percentage of packets to drop on a random basis.

---

### Default

No packets will be dropped (i.e. 0%).

---

### Usage Guidelines

Use this command to set the percentage of packets to drop in a network emulator QoS policy. This emulates link problems which cause packet loss.

Use the **set** form of this command to specify the percentage of packets to randomly drop for a network emulator QoS policy.

Use the **delete** form of this command to restore the default percentage of packets to drop for a network emulator QoS policy.

Use the **show** form of this command to display packet loss configuration.

## traffic-policy network-emulator <policy-name> packet-reordering <percent>

Sets the percentage of packets to reorder in a network emulation QoS policy.

---

### Syntax

```
set traffic-policy network-emulator policy-name packet-reordering percent[%]  
delete traffic-policy network-emulator policy-name packet-reordering  
show traffic-policy network-emulator policy-name packet-reordering
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    network-emulator policy-name {  
        packet-reordering percent  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the network emulator policy.
<i>percent</i>	The percentage of packets to reorder on a random basis.

---

### Default

No packets will be reordered (i.e. 0%).

---

### Usage Guidelines

Use this command to set the percentage of packets to reorder in a network emulator QoS policy. This emulates network issues which cause packet reordering. This mechanism will only have an impact when more than one packet is in the queue.

Use the set form of this command to specify the percentage of packets to randomly reorder for a network emulator QoS policy.



Use the **delete** form of this command to restore the default percentage of packets to reorder for a network emulator QoS policy.

Use the **show** form of this command to display packet reordering configuration.

## traffic-policy network-emulator <policy-name> queue-limit <limit>

Sets an upper bound for the number of packets allowed in the queue for a network emulation QoS policy.

---

### Syntax

```
set traffic-policy network-emulator policy-name queue-limit limit  
delete traffic-policy network-emulator policy-name queue-limit  
show traffic-policy network-emulator policy-name queue-limit
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    network-emulator policy-name {  
        queue-limit limit  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the network emulator policy.
<i>limit</i>	Mandatory. The maximum queue size, in packets. The range is 0 to 4294967295. The default is 127.

---

### Default

A queue is not permitted to exceed 127 packets.

---

### Usage Guidelines

Use this command to set the maximum number of packets that can wait in a queue for this queuing policy. If maximum queue size is reached, the system begins dropping packets.

Use the **set** form of this command to set the queue limit.

Use the **delete** form of this command to restore the default queue limit.

Use the **show** form of this command to display queue limit configuration.

## traffic-policy priority-queue <policy-name>

Defines a Priority Queue QoS policy.

---

### Syntax

```
set traffic-policy priority-queue policy-name
delete traffic-policy priority-queue policy-name
show traffic-policy priority-queue policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    priority-queue policy-name {
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
--------------------	---

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a priority queue QoS policy. The priority queue policy acts on outbound traffic only. The policy name must be unique and not used with other QoS policy commands.

The priority queue mechanism provides up to seven queues, each with a different priority. Packets are placed in the queues based on match criteria associated with each queue. Packets are retrieved from the queues in priority order. Packets in lower priority queues will not be transmitted until those in higher priority queues have been sent. Individual queues can be configured as drop-tail, fair-queue, or random-detect.

Use the **set** form of this command to create a priority queue QoS policy.

Use the **delete** form of this command to remove a priority queue QoS policy.

Use the **show** form of this command to display priority queue QoS policy configuration.

## traffic-policy priority-queue <policy-name> class <class>

Defines a traffic class for a round robin QoS policy.

---

### Syntax

```
set traffic-policy priority-queue policy-name class class
delete traffic-policy priority-queue policy-name class class
show traffic-policy priority-queue policy-name class class
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    priority-queue policy-name {
        class class {
        }
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a traffic class for a priority queue QoS policy. Each class is a separate queue, with class 1 being the lowest priority and class 7 being the highest priority. Packets are retrieved from higher priority queues before they are retrieved from lower priority queues.

Use the set form of this command to create a traffic class in a priority queue QoS policy.

Use the **delete** form of this command to remove a traffic class from a priority queue QoS policy.

Use the **show** form of this command to display traffic class configuration within a priority queue QoS policy.

## traffic-policy priority-queue <policy-name> class <class> description <desc>

Sets a description for a traffic class.

---

### Syntax

```
set traffic-policy priority-queue policy-name class class description desc  
delete traffic-policy priority-queue policy-name class class description  
show traffic-policy priority-queue policy-name class class description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    priority-queue policy-name {  
        class class {  
            description desc  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>desc</i>	The description for this traffic class.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic class.

Use the **set** form of this command to specify a description.



Use the **delete** form of this command to remove a description.

Use the **show** form of this command to display description configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name>

Defines a traffic class matching rule.

---

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name  
delete traffic-policy priority-queue policy-name class class match match-name  
show traffic-policy priority-queue policy-name class class match match-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    priority-queue policy-name {  
        class class {  
            match match-name {  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.

---

### Default

None.

---

### Usage Guidelines

Use this command to define a rule setting out the match conditions for membership in a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to create the traffic class matching rule.

Use the **delete** form of this command to remove the traffic class matching rule configuration node.

Use the **show** form of this command to display traffic class matching rule configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> description <desc>

Sets a description for a match rule.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name
description desc
```

```
delete traffic-policy priority-queue policy-name class class match match-name
description
```

```
show traffic-policy priority-queue policy-name class class match match-name
description
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    priority-queue policy-name {
        class class {
            match match-name {
                description desc
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>desc</i>	The description for this match.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic class matching rule.

Use the **set** form of this command to set the description.

Use the **delete** form of this command to remove the description.

Use the **show** form of this command to display description configuration.

## **traffic-policy priority-queue <policy-name> class <class> match <match-name> ether destination <mac-addr>**

Specifies a match criterion based on Ethernet destination (MAC) address.

---

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ether  
destination mac-addr
```

```
delete traffic-policy priority-queue policy-name class class match match-name ether  
destination
```

```
show traffic-policy priority-queue policy-name class class match match-name ether  
destination
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    priority-queue policy-name {  
        class class {  
            match match-name {  
                ether {  
                    destination mac-addr  
                }  
            }  
        }  
    }  
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>mac-addr</i>	Performs a match based on the destination MAC address on the interface the policy is applied to. The format is 6 colon-separated 8-bit numbers in hexadecimal; for example, 00:0a:59:9a:f2:ba.

---

---

### Default

If not set, packets are not matched against the destination MAC address.

---

### Usage Guidelines

Use this command to define a match condition based on destination MAC address for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> ether protocol <num>

Specifies a match criterion based on Ethernet packet type.

---

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ether  
protocol num  
delete traffic-policy priority-queue policy-name class class match match-name ether  
protocol  
show traffic-policy priority-queue policy-name class class match match-name ether  
protocol
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    priority-queue policy-name {  
        class class {  
            match match-name {  
                ether {  
                    protocol num  
                }  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>num</i>	Performs a match based on the Ethernet packet type (i.e. protocol number). The range is 0 to 65535.



---

### Default

If not set, packets are not matched against the Ethernet packet type.

---

### Usage Guidelines

Use this command to define a match condition based on Ethernet packet type for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify the packet type to be matched.

Use the **delete** form of this command to remove packet type as a match condition.

Use the **show** form of this command to display packet type match condition configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> ether source <mac-addr>

Specifies a match criterion based on Ethernet source (MAC) address.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ether
source mac-addr

delete traffic-policy priority-queue policy-name class class match match-name ether
source

show traffic-policy priority-queue policy-name class class match match-name ether
source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  priority-queue policy-name {
    class class {
      match match-name {
        ether {
          source mac-addr
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>mac-addr</i>	Performs a match based on the MAC address of the interface the policy is applied to. The format is 6 colon-separated 8-bit numbers in hexadecimal; for example, 00:0a:59:9a:f2:ba.

---

### Default

If not set, packets are not matched against the source MAC address.

---

### Usage Guidelines

Use this command to define a match condition based on source MAC address for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> interface <interface>

Specifies a match criterion based on incoming interface.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name interface interface
```

```
delete traffic-policy priority-queue policy-name class class match match-name interface
```

```
show traffic-policy priority-queue policy-name class class match match-name interface
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    priority-queue policy-name {
        class class {
            match match-name {
                interface interface
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>interface</i>	Performs a match based on the specified Ethernet interface name. The ingress interface for incoming traffic will be compared with this value.

---

## Default

None.

---

## Usage Guidelines

Use this command to set a match condition for a traffic class based on incoming interface.

If incoming packets ingress through the interface specified by this command, the traffic is a member of this traffic class (provided other match conditions are satisfied).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify an interface to be matched by incoming packets.

Use the **delete** form of this command to remove the match interface.

Use the **show** form of this command to display interface match configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> ip destination

Specifies a match criterion based on IP destination information.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ip
destination {address ipv4net | port port}
```

```
delete traffic-policy priority-queue policy-name class class match match-name ip
destination [address | port]
```

```
show traffic-policy priority-queue policy-name class class match match-name ip
destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  priority-queue policy-name {
    class class {
      match match-name {
        ip {
          destination {
            address ipv4net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.

---

<i>ipv4net</i>	Performs a match based on the destination IP subnet address.
<i>port</i>	Performs a match based on destination port. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.

---

---

### Default

If not set, packets are not matched against destination information.

---

### Usage Guidelines

Use this command to define a match condition based on destination subnet address and/or port for a traffic class.

You can match packets based on a destination represented by either or both of IP subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## **traffic-policy priority-queue <policy-name> class <class> match <match-name> ip dscp <value>**

Specifies a match criterion based on the value of the DSCP field.

---

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ip dscp  
value
```

```
delete traffic-policy priority-queue policy-name class class match match-name ip  
dscp
```

```
show traffic-policy priority-queue policy-name class class match match-name ip dscp
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    priority-queue policy-name {  
        class class {  
            match match-name {  
                ip {  
                    dscp value  
                }  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.



---

<i>value</i>	Performs a match based on the specified value. This value is compared with the value in the DSCP field of the ToS byte in the IP header. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <code>/etc/iproute2/route_dscp</code> (for example, <code>lowdelay</code> ).
--------------	--

---

---

### Default

If not set, packets are not matched against DSCP value.

---

### Usage Guidelines

Use this command to define a match condition based on the Differentiated Services Code Point (DSCP) field.

The DSCP field is a 6-bit field in the Type of Service (ToS) byte of the IP header. It provides a way of marking packets in order to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on DSCP value.

Use the **delete** form of this command to remove DSCP as a match condition.

Use the **show** form of this command to display DSCP value configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> ip protocol <proto>

Specifies a match criterion based on the IP protocol.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ip
protocol proto
delete traffic-policy priority-queue policy-name class class match match-name ip
protocol
show traffic-policy priority-queue policy-name class class match match-name ip
protocol
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    priority-queue policy-name {
        class class {
            match match-name {
                ip {
                    protocol proto
                }
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>proto</i>	Performs a match based on the protocol name (for example, <b>icmp</b> ) or number, as assigned by the IANA.

---

### Default

If not set, packets are not matched against IP protocol.

---

### Usage Guidelines

Use this command to define a match condition for a traffic class based on protocol.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on protocol.

Use the **delete** form of this command to remove protocol value as a match condition.

Use the **show** form of this command to match condition protocol configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> ip source

Specifies a match criterion based on source IP information.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ip source
{address ipv4net | port port}
```

```
delete traffic-policy priority-queue policy-name class class match match-name ip
source {address | port}
```

```
show traffic-policy priority-queue policy-name class class match match-name ip
source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  priority-queue policy-name {
    class class {
      match match-name {
        ip {
          source {
            address ipv4net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.

---

<i>ipv4net</i>	The source IP subnet address to match for this rule.
<i>port</i>	The source port to match for this rule. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.

---

---

### Default

If not set, packets are not matched against IP source information.

---

### Usage Guidelines

Use this command to define a match condition based on source subnet address and/or port for a traffic class.

You can match packets based on a source represented by either or both of IP subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 destination

Specifies a match criterion based on IPv6 destination information.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ipv6
destination {address ipv6net | port port}
```

```
delete traffic-policy priority-queue policy-name class class match match-name ipv6
destination [address | port]
```

```
show traffic-policy priority-queue policy-name class class match match-name ipv6
destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  priority-queue policy-name {
    class class {
      match match-name {
        ipv6 {
          destination {
            address ipv6net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority-queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.

---

<i>ipv6net</i>	Performs a match based on the destination IPv6 subnet address.
<i>port</i>	Performs a match based on destination port. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.

---

---

### Default

If not set, packets are not matched against destination information.

---

### Usage Guidelines

Use this command to define a match condition based on destination subnet address and/or port for a traffic class.

You can match packets based on a destination represented by either or both of IPv6 subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## **traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 dscp <value>**

Specifies a match criterion based on the value of the DSCP field.

---

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ipv6 dscp  
value
```

```
delete traffic-policy priority-queue policy-name class class match match-name ipv6  
dscp
```

```
show traffic-policy priority-queue policy-name class class match match-name ipv6  
dscp
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    priority-queue policy-name {  
        class class {  
            match match-name {  
                ipv6 {  
                    dscp value  
                }  
            }  
        }  
    }  
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.

---



---

<i>value</i>	Performs a match based on the specified value. This value is compared with the value in the DSCP field of the ToS byte in the IP header. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <i>/etc/iproute2/route_dsfld</i> (for example, <b>lowdelay</b> ).
--------------	---

---

---

### Default

If not set, packets are not matched against DSCP value.

---

### Usage Guidelines

Use this command to define a match condition based on the Differentiated Services Code Point (DSCP) field.

The DSCP field is a 6-bit field in the Type of Service (ToS) byte of the IP header. It provides a way of marking packets in order to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on DSCP value.

Use the **delete** form of this command to remove DSCP as a match condition.

Use the **show** form of this command to display DSCP value configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 protocol <proto>

Specifies a match criterion based on the IPv6 protocol.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ipv6
protocol proto
```

```
delete traffic-policy priority-queue policy-name class class match match-name ipv6
protocol
```

```
show traffic-policy priority-queue policy-name class class match match-name ipv6
protocol
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  priority-queue policy-name {
    class class {
      match match-name {
        ipv6 {
          protocol proto
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>proto</i>	Performs a match based on the protocol name (for example, <b>icmp</b> ) or number, as assigned by the IANA.

---

### Default

If not set, packets are not matched against IP protocol.

---

### Usage Guidelines

Use this command to define a match condition for a traffic class based on protocol.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on protocol.

Use the **delete** form of this command to remove protocol value as a match condition.

Use the **show** form of this command to match condition protocol configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> ipv6 source

Specifies a match criterion based on source IPv6 information.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name ipv6
source {address ipv6net | port port}
```

```
delete traffic-policy priority-queue policy-name class class match match-name ipv6
source {address | port}
```

```
show traffic-policy priority-queue policy-name class class match match-name ipv6
source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  priority-queue policy-name {
    class class {
      match match-name {
        ipv6 {
          source {
            address ipv6net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 3 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.

---

<i>ipv6net</i>	The source IPv6 subnet address to match for this rule.
<i>port</i>	The source port to match for this rule. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.

---

---

### Default

If not set, packets are not matched against IPv6 source information.

---

### Usage Guidelines

Use this command to define a match condition based on source subnet address and/or port for a traffic class.

You can match packets based on a source represented by either or both of IPv6 subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> mark <fwmark>

Specifies a match criterion based on the value of the fwmark field.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name mark  
fwmark
```

```
delete traffic-policy priority-queue policy-name class class match match-name mark
```

```
show traffic-policy priority-queue policy-name class class match match-name mark
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {  
    priority-queue policy-name {  
        class class {  
            match match-name {  
                mark fwmark  
            }  
        }  
    }  
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority-queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>fwmark</i>	Performs a match based on the fwmark field. The range is 0 to 4294967295.

### Default

If not set, packets are not matched against fwmark.

---

### Usage Guidelines

Use this command to define a match condition based on the fwmark field for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a fwmark value to be matched.

Use the **delete** form of this command to remove mark as a match condition.

Use the **show** form of this command to display the mark match condition configuration.

## traffic-policy priority-queue <policy-name> class <class> match <match-name> vif <vlan-id>

Specifies a match criterion based on VLAN ID.

### Syntax

```
set traffic-policy priority-queue policy-name class class match match-name vif
vlan-id
```

```
delete traffic-policy priority-queue policy-name class class match match-name vif
```

```
show traffic-policy priority-queue policy-name class class match match-name vif
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    priority-queue policy-name {
        class class {
            match match-name {
                vif vlan-id
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority-queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>vlan-id</i>	Performs a match based on VLAN ID. The range is 1 to 4096.

### Default

If not set, packets are not matched against VLAN ID.



---

### Usage Guidelines

Use this command to define a match condition based on VLAN ID for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a VLAN ID to be matched.

Use the **delete** form of this command to remove VLAN ID as a match condition.

Use the **show** form of this command to display VLAN ID match condition configuration.

## traffic-policy priority-queue <policy-name> class <class> queue-limit <limit>

Specifies the maximum queue size for a traffic class.

---

### Syntax

```
set traffic-policy priority-queue policy-name class class queue-limit limit  
delete traffic-policy priority-queue policy-name class class queue-limit  
show traffic-policy priority-queue policy-name class class queue-limit
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    priority-queue policy-name {  
        class class {  
            queue-limit limit  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>limit</i>	The maximum queue size in packets.

---

### Default

The default maximum queue size is 1000 packets.

---

### Usage Guidelines

Use this command to set the maximum queue size (in packets) for a traffic class.  
Use the **set** form of this command to specify the queue limit.

Use the **delete** form of this command to remove queue limit.

Use the **show** form of this command to display queue limit configuration.

## traffic-policy priority-queue <policy-name> class <class> queue-type <type>

Specifies the type of queuing to use for a traffic class.

### Syntax

```
set traffic-policy priority-queue policy-name class class queue-type type
delete traffic-policy priority-queue policy-name class class queue-type
show traffic-policy priority-queue policy-name class class queue-type
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    priority-queue policy-name {
        class class {
            queue-type type
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>class</i>	Mandatory. The class ID. The range is 1 to 7.
<i>type</i>	The queuing method to use. Supported values are as follows: <b>drop-tail:</b> Uses a First In First Out (FIFO) queue. <b>fair-queue:</b> Uses a Stochastic Fair Queue (SFQ) queue. <b>random-detect:</b> Uses a Random Early Detection (RED) queue.

### Default

The default is **drop-tail**.

---

### Usage Guidelines

Use this command to set the type of queuing mechanism to use for a traffic class.

Use the **set** form of this command to specify the queue type.

Use the **delete** form of this command to restore the default queue type.

Use the **show** form of this command to display queue type configuration.

## traffic-policy priority-queue <policy-name> default

Defines a default priority queue QoS policy.

---

### Syntax

```
set traffic-policy priority-queue policy-name default
delete traffic-policy priority-queue policy-name default
show traffic-policy priority-queue policy-name default
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    priority-queue policy-name {
        default {
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority-queue policy.
--------------------	---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a default priority queue policy. This policy will be applied to all traffic that does not match any other defined class.

Use the **set** form of this command to create the default class configuration node.

Use the **delete** form of this command to remove the default class configuration node.

Use the **show** form of this command to display the default class configuration node.

## traffic-policy priority-queue <policy-name> default queue-limit <limit>

Specifies the maximum queue size for the default traffic class.

---

### Syntax

```
set traffic-policy priority-queue policy-name default queue-limit limit
delete traffic-policy priority-queue policy-name default queue-limit
show traffic-policy priority-queue policy-name default queue-limit
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  priority-queue policy-name {
    default {
      queue-limit limit
    }
  }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>limit</i>	The maximum queue size in packets.

---

### Default

The default maximum queue size is 1000 packets.

---

### Usage Guidelines

Use this command to set the maximum queue size (in packets) for the default class.  
Use the **set** form of this command to specify the queue limit.  
Use the **delete** form of this command to remove queue limit.

Use the **show** form of this command to display queue limit configuration.



## traffic-policy priority-queue <policy-name> default queue-type <type>

Specifies the type of queuing to use for the default traffic class.

---

### Syntax

```
set traffic-policy priority-queue policy-name default queue-type type
delete traffic-policy priority-queue policy-name default queue-type
show traffic-policy priority-queue policy-name default queue-type
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  priority-queue policy-name {
    default {
      queue-type type
    }
  }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the priority-queue policy.
<i>type</i>	The queuing method to use. Supported values are as follows: <b>drop-tail:</b> Uses a First In First Out (FIFO) queue. <b>fair-queue:</b> Uses a Stochastic Fair Queue (SFQ) queue. <b>random-detect:</b> Uses a Random Early Detection (RED) queue.

---

---

### Default

The default is **drop-tail**.

---

### Usage Guidelines

Use this command to set the type of queuing mechanism to use for the default traffic class.

Use the **set** form of this command to specify the queue type.

Use the **delete** form of this command to restore the default queue type.

Use the **show** form of this command to display queue type configuration.

## traffic-policy priority-queue <policy-name> description <desc>

Specifies a description for a priority-queue QoS policy.

---

### Syntax

```
set traffic-policy priority-queue policy-name description desc
delete traffic-policy priority-queue policy-name description
show traffic-policy priority-queue policy-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    priority-queue policy-name {
        description desc
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the priority queue policy.
<i>desc</i>	The description for this priority queue policy.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a priority queue policy.

Use the **set** form of this command to specify a description for a priority queue policy.

Use the **delete** form of this command to remove a description from a priority queue policy.

Use the **show** form of this command to display description configuration for a priority queue policy.

## traffic-policy random-detect <policy-name>

Defines a Weighted Random Early Detection (WRED) QoS policy.

---

### Syntax

```
set traffic-policy random-detect policy-name
delete traffic-policy random-detect policy-name
show traffic-policy random-detect policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    random-detect policy-name {
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the random detect policy.
--------------------	--

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a random detect QoS policy based on the Weighted Random Early Detection (WRED) congestion avoidance mechanism. Random detect policy acts on outbound traffic only. The policy name must be unique and not used with other QoS policy commands.

The Random Early Detection (RED) mechanism randomly drops packets prior to periods of high congestion in order to signal the source to decrease its transmission rate. This helps avoid the condition where output buffers fill and packets at the tail of the buffer (as well as newly arriving packets to the buffer) are dropped. This can cause global resynchronization of TCP hosts as multiple hosts reduce their transmission rates. Then, as the congestion clears, the transmission rates are increased to the point where congestion reoccurs. This cycle of congestion and

congestion clearing does not make the best use of the available bandwidth. RED reduces the chance that this issue will occur by selectively dropping packets when the output interface shows signs of congestion. This in turn reduces the chance of global synchronization and makes better use of available bandwidth.

WRED takes RED one step further by providing a way to attach precedence to different traffic streams and hence provide different quality of service to different traffic by dropping more packets from certain traffic streams than from others.

Use the **set** form of this command to create a random detect QoS policy.

Use the **delete** form of this command to remove a random detect QoS policy.

Use the **show** form of this command to display random detect QoS policy configuration.

## traffic-policy random-detect <policy-name> bandwidth

Specifies the bandwidth limit for all combined traffic constrained by this policy.

---

### Syntax

```
set traffic-policy random-detect policy-name bandwidth [auto | rate | rate-suffix]  
delete traffic-policy random-detect policy-name bandwidth  
show traffic-policy random-detect policy-name bandwidth
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    random-detect policy-name {  
        bandwidth [auto | rate | rate-suffix]  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the random detect policy.
<i>auto</i>	The bandwidth is based on the speed of the interface. This is the default.
<i>rate</i>	The bandwidth, specified in kilobits per second.
<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported: <b>kbit</b> : Kilobits per second. <b>mbit</b> : Megabits per second. <b>gbit</b> : Gigabits per second. <b>kbps</b> : Kilobytes per second. <b>mbps</b> : Megabytes per second. <b>gbps</b> : Gigabytes per second.

---

---

### Default

The bandwidth is based on the interface that the policy is applied to.

---

### Usage Guidelines

Use this command to set bandwidth constraints for a random detect QoS policy. This is the maximum bandwidth available for all classes.

Use the **set** form of this command to specify bandwidth constraints for the policy.

Use the **delete** form of this command to restore default bandwidth constraints for the policy.

Use the **show** form of this command to display policy bandwidth configuration.



## traffic-policy random-detect <policy-name> description <desc>

Sets a description for a random-detect policy.

---

### Syntax

```
set traffic-policy random-detect policy-name description desc  
delete traffic-policy random-detect policy-name description  
show traffic-policy random-detect policy-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    random-detect policy-name {  
        description desc  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the random detect policy.
<i>desc</i>	The description for this random detect policy.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a random detect policy.

Use the **set** form of this command to specify a description.

Use the **delete** form of this command to remove a description.

Use the **show** form of this command to display description configuration.

## traffic-policy random-detect <policy-name> precedence <precedence>

Sets parameters for dropping packets based on precedence for a random-detect policy.

### Syntax

```
set traffic-policy random-detect policy-name precedence precedence [average-packet
bytes | mark-probability probability | max-threshold max | min-probability min |
queue-limit packets]

delete traffic-policy random-detect policy-name precedence precedence
[average-packet | mark-probability | max-threshold | min-probability | queue-limit]

show traffic-policy random-detect policy-name precedence precedence
[average-packet | mark-probability | max-threshold | min-probability | queue-limit]
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    random-detect policy-name {
        precedence precedence {
            average-packet bytes
            mark-probability probability
            max-threshold max
            min-threshold min
            queue-limit packets
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the random detect policy.
<i>bytes</i>	The average packet size in bytes. The default is 1024.
<i>precedence</i>	The IP precedence (the first three bits of the TOS field) of the packet.

<i>probability</i>	The fraction of packets (i.e. 1/probability) dropped when the average queue depth reaches the maximum threshold. The default is 10.
<i>max</i>	All packets are dropped when the average queue depth goes above this threshold. The range is 0 to 4096 packets. The default is 18.
<i>min</i>	<p>Packets begin to be dropped when the average queue depth reaches this threshold. The range is 0 to 4096 packets. The default depends on the <i>precedence</i>:</p> <p>Precedence 0 -&gt; min-threshold = 9</p> <p>Precedence 1-&gt; min-threshold = 10</p> <p>Precedence 2 -&gt; min-threshold = 11</p> <p>Precedence 3 -&gt; min-threshold = 12</p> <p>Precedence 4 -&gt; min-threshold = 13</p> <p>Precedence 5 -&gt; min-threshold = 14</p> <p>Precedence 6 -&gt; min-threshold = 15</p> <p>Precedence 7 -&gt; min-threshold = 16</p>
<i>packets</i>	All packets are dropped when the instantaneous queue depth reaches this threshold. The default is 4 * <b>max-threshold</b> .

### Default

None.

### Usage Guidelines

This feature uses the first three bits of the Type of Service (TOS) field to categorize data streams. Within each of these streams parameters can be set to adjust the rate that packets are dropped when congestion occurs. Each time a packet arrives and is to be sent out the interface a decision is made based on the packet precedence and the parameters set for the specified precedence. If the average output queue size is less than the **min-threshold** then the packet is placed on the output queue. If the average output queue size is between the **min-threshold** and the **max-threshold** the packet may be queued or dropped based on the **probability**. When the average output queue size is larger than **max-threshold** all packets are dropped. When the instantaneous queue size is larger than **queue-limit** all packets are dropped.

If **max-threshold** is set and **min-threshold** is not set then **min-threshold** is automatically scaled to 1/2 **max-threshold**. In addition, the system enforces the constraint that: **min-threshold** < **max-threshold** < **queue-limit**.

**NOTE** *Non-IP packets are treated as precedence 0.*

Use this command to specify the packet dropping parameters for a random detect policy.

Use the **set** form of this command to specify the packet dropping parameters for a random detect policy.

Use the **delete** form of this command to remove the packet dropping parameters for a random detect policy.

Use the **show** form of this command to display the packet dropping parameters for a random detect policy.

## traffic-policy rate-control <policy-name>

Defines a rate controlling QoS policy.

---

### Syntax

```
set traffic-policy rate-control policy-name
delete traffic-policy rate-control policy-name
show traffic-policy rate-control policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    rate-control policy-name {
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the rate controlling policy.
--------------------	---

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a rate controlling QoS policy. Rate control policy acts on outbound traffic only. The policy name must be unique and not used with other QoS policy commands.

The Vyatta system uses a version of the Token Bucket Filter (TBF) algorithm. TBF is a classless queuing discipline that only passes packets arriving at a rate which is not exceeding some administratively set rate, but with the possibility to allow short bursts in excess of this rate.

Use the **set** form of this command to create a rate controlling QoS policy.

Use the **delete** form of this command to remove a rate controlling QoS policy.

Use the **show** form of this command to display rate controlling QoS policy configuration.

## traffic-policy rate-control <policy-name> bandwidth

Specifies the bandwidth limit for all combined traffic constrained by this policy.

---

### Syntax

```
set traffic-policy rate-control policy-name bandwidth [rate | rate-suffix]  
delete traffic-policy rate-control policy-name bandwidth  
show traffic-policy rate-control policy-name bandwidth
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    rate-control policy-name {  
        bandwidth [rate | rate-suffix]  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the rate controlling policy.
<i>rate</i>	The bandwidth, specified in kilobits per second.
<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported:  kbit: Kilobits per second. mbit: Megabits per second. gbit: Gigabits per second. kbits: Kilobytes per second. mbps: Megabytes per second. gbps: Gigabytes per second.

---

### Default

None.

---

### Usage Guidelines

Use this command to set bandwidth constraints for a rate controlling QoS policy. This is the maximum bandwidth available for all classes and must be set.

Use the **set** form of this command to specify bandwidth constraints for the policy.

Use the **delete** form of this command to restore default bandwidth constraints for the policy.

Use the **show** form of this command to display policy bandwidth configuration.



## traffic-policy rate-control <policy-name> burst

Sets the burst size for a rate controlling QoS policy.

---

### Syntax

```
set traffic-policy rate-control policy-name burst [num | num-suffix]  
delete traffic-policy rate-control policy-name burst  
show traffic-policy rate-control policy-name burst
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    rate-control policy-name {  
        burst [num | num-suffix]  
    }  
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the rate controlling policy.
<i>num</i>	The burst size, specified in bytes.
<i>num-suffix</i>	The burst size, specified as a number and a scaling suffix (for example, 10mb). The following suffixes are supported:  kb: Kilobytes. mb: Megabytes. gb: Gigabytes.

---

---

### Default

The default burst size is 15 kilobytes.

---

### Usage Guidelines

Use this command to set the burst size for a rate controlling QoS policy. This is the maximum amount of traffic that may be sent at a given time.

Use the **set** form of this command to specify the burst size for a rate controlling QoS policy.

Use the **delete** form of this command to restore the default burst size for a rate controlling QoS policy.

Use the **show** form of this command to display rate control burst size configuration.

## traffic-policy rate-control <policy-name> description <desc>

Sets a description for a rate controlling policy.

---

### Syntax

```
set traffic-policy rate-control policy-name description desc  
delete traffic-policy rate-control policy-name description  
show traffic-policy rate-control policy-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    rate-control policy-name {  
        description desc  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the rate control policy.
<i>desc</i>	The description for this rate control policy.

---

### Default

None.

---

### Usage Guidelines

- Use this command to record a description for a rate control policy.
- Use the **set** form of this command to specify a description.
- Use the **delete** form of this command to remove a description.
- Use the **show** form of this command to display description configuration.

## traffic-policy rate-control <policy-name> latency

Sets the limit on queue size based on latency for a rate controlling QoS policy.

---

### Syntax

```
set traffic-policy rate-control policy-name latency [num | num-suffix]
delete traffic-policy rate-control policy-name latency
show traffic-policy rate-control policy-name latency
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    rate-control policy-name {
        latency [num | num-suffix]
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the rate controlling policy.
<i>num</i>	The latency, specified in milliseconds.
<i>num-suffix</i>	The latency, specified as a time and a scaling suffix (for example, 10ms). The following suffixes are supported:  secs: Seconds. ms: Milliseconds. us: Microseconds.

---

---

### Default

The default latency is 50 milliseconds.

---

### Usage Guidelines

Use this command to set the latency for a rate controlling QoS policy. This is the maximum amount of time a packet can sit in the Token Bucket Filter.

Use the **set** form of this command to specify the latency for a rate controlling QoS policy.

Use the **delete** form of this command to restore the default latency for a rate controlling QoS policy.

Use the **show** form of this command to display rate control latency configuration.

## traffic-policy round-robin <policy-name>

Defines a round robin QoS policy.

---

### Syntax

```
set traffic-policy round-robin policy-name
delete traffic-policy round-robin policy-name
show traffic-policy round-robin policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    round-robin policy-name {
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the round robin policy.
--------------------	--

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a round robin QoS policy. Round robin policy acts on outbound traffic only. The policy name must be unique and not used with other QoS policy commands. The round robin policy provides round-robin fairness to all classes. The difference between **shaper** and **round-robin** is that **shaper** limits bandwidth usage by class and then allocates any leftover bandwidth. **Round-robin**, on the other hand, attempts to divide all available bandwidth between the defined classes.

Use the **set** form of this command to create a round robin QoS policy.

Use the **delete** form of this command to remove a round robin QoS policy.

Use the **show** form of this command to display round robin QoS policy configuration.

## traffic-policy round-robin <policy-name> class <class>

Defines a traffic class for a round robin QoS policy.

---

### Syntax

```
set traffic-policy round-robin policy-name class class
delete traffic-policy round-robin policy-name class class
show traffic-policy round-robin policy-name class class
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    round-robin policy-name {
        class class {
        }
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a traffic class for a round robin QoS policy. This allows packets to be grouped into various traffic classes, which can be treated with different levels of service.

Use the **set** form of this command to create a traffic class in a round robin QoS policy.

Use the **delete** form of this command to remove a traffic class from a round robin QoS policy.



Use the **show** form of this command to display traffic class configuration within a round robin QoS policy.

## traffic-policy round-robin <policy-name> class <class> description <desc>

Sets a description for a traffic class.

---

### Syntax

```
set traffic-policy round-robin policy-name class class description desc  
delete traffic-policy round-robin policy-name class class description  
show traffic-policy round-robin policy-name class class description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    round-robin policy-name {  
        class class {  
            description desc  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>desc</i>	The description for this traffic class.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic class.

Use the **set** form of this command to specify a description.

Use the **delete** form of this command to remove a description.

Use the **show** form of this command to display description configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name>

Defines a traffic class matching rule.

---

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name  
delete traffic-policy round-robin policy-name class class match match-name  
show traffic-policy round-robin policy-name class class match match-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    round-robin policy-name {  
        class class {  
            match match-name {  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.

---

### Default

None.

---

### Usage Guidelines

Use this command to define a rule setting out the match conditions for membership in a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to create the traffic class matching rule.

Use the **delete** form of this command to remove the traffic class matching rule configuration node.

Use the **show** form of this command to display traffic class matching rule configuration.

## **traffic-policy round-robin <policy-name> class <class> match <match-name> description <desc>**

Sets a description for a match rule.

---

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name description  
desc
```

```
delete traffic-policy round-robin policy-name class class match match-name  
description
```

```
show traffic-policy round-robin policy-name class class match match-name  
description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    round-robin policy-name {  
        class class {  
            match match-name {  
                description desc  
            }  
        }  
    }  
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>desc</i>	The description for this match.

---

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic class matching rule.

Use the **set** form of this command to set the description.

Use the **delete** form of this command to remove the description.

Use the **show** form of this command to display description configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ether destination <mac-addr>

Specifies a match criterion based on Ethernet destination (MAC) address.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ether
destination mac-addr
```

```
delete traffic-policy round-robin policy-name class class match match-name ether
destination
```

```
show traffic-policy round-robin policy-name class class match match-name ether
destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        ether {
          destination mac-addr
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>mac-addr</i>	Performs a match based on the destination MAC address on the interface the policy is applied to. The format is 6 colon-separated 8-bit numbers in hexadecimal; for example, 00:0a:59:9a:f2:ba.



---

### Default

If not set, packets are not matched against the destination MAC address.

---

### Usage Guidelines

Use this command to define a match condition based on destination MAC address for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ether protocol <num>

Specifies a match criterion based on Ethernet packet type.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ether
protocol num
```

```
delete traffic-policy round-robin policy-name class class match match-name ether
protocol
```

```
show traffic-policy round-robin policy-name class class match match-name ether
protocol
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        ether {
          protocol num
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>num</i>	Performs a match based on the Ethernet packet type (i.e. protocol number). The range is 0 to 65535.

---

### Default

If not set, packets are not matched against the Ethernet packet type.

---

### Usage Guidelines

Use this command to define a match condition based on Ethernet packet type for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify the packet type to be matched.

Use the **delete** form of this command to remove packet type as a match condition.

Use the **show** form of this command to display packet type match condition configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ether source <mac-addr>

Specifies a match criterion based on Ethernet source (MAC) address.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ether
source mac-addr
```

```
delete traffic-policy round-robin policy-name class class match match-name ether
source
```

```
show traffic-policy round-robin policy-name class class match match-name ether
source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        ether {
          source mac-addr
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>mac-addr</i>	Performs a match based on the MAC address of the interface the policy is applied to. The format is 6 colon-separated 8-bit numbers in hexadecimal; for example, 00:0a:59:9a:f2:ba.

---

### Default

If not set, packets are not matched against the source MAC address.

---

### Usage Guidelines

Use this command to define a match condition based on source MAC address for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> interface <interface>

Specifies a match criterion based on incoming interface.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name interface
interface
```

```
delete traffic-policy round-robin policy-name class class match match-name interface
```

```
show traffic-policy round-robin policy-name class class match match-name interface
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    round-robin policy-name {
        class class {
            match match-name {
                interface interface
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>interface</i>	Performs a match based on the specified Ethernet interface name. The ingress interface for incoming traffic will be compared with this value.

---

## Default

None.

---

## Usage Guidelines

Use this command to set a match condition for a traffic class based on incoming interface.

If incoming packets ingress through the interface specified by this command, the traffic is a member of this traffic class (provided other match conditions are satisfied).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify an interface to be matched by incoming packets.

Use the **delete** form of this command to remove the match interface.

Use the **show** form of this command to display interface match configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ip destination

Specifies a match criterion based on IP destination information.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ip
destination {address ipv4net | port port}

delete traffic-policy round-robin policy-name class class match match-name ip
destination [address | port]

show traffic-policy round-robin policy-name class class match match-name ip
destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        ip {
          destination {
            address ipv4net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.



---

<i>ipv4net</i>	Performs a match based on the destination IP subnet address.
<i>port</i>	Performs a match based on destination port. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.

---

---

### Default

If not set, packets are not matched against destination information.

---

### Usage Guidelines

Use this command to define a match condition based on destination subnet address and/or port for a traffic class.

You can match packets based on a destination represented by either or both of IP subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a destination to be matched.-

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ip dscp <value>

Specifies a match criterion based on the value of the DSCP field.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ip dscp
value
```

```
delete traffic-policy round-robin policy-name class class match match-name ip dscp
```

```
show traffic-policy round-robin policy-name class class match match-name ip dscp
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        ip {
          dscp value
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.

---

<i>value</i>	Performs a match based on the specified value. This value is compared with the value in the DSCP field of the ToS byte in the IP header. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <i>/etc/iproute2/route_dsfld</i> (for example, <b>lowdelay</b> ).
--------------	---

---

---

### Default

If not set, packets are not matched against DSCP value.

---

### Usage Guidelines

Use this command to define a match condition based on the Differentiated Services Code Point (DSCP) field.

The DSCP field is a 6-bit field in the Type of Service (ToS) byte of the IP header. It provides a way of marking packets in order to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on DSCP value.

Use the **delete** form of this command to remove DSCP as a match condition.

Use the **show** form of this command to display DSCP value configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ip protocol <proto>

Specifies a match criterion based on the IP protocol.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ip protocol
proto
```

```
delete traffic-policy round-robin policy-name class class match match-name ip
protocol
```

```
show traffic-policy round-robin policy-name class class match match-name ip
protocol
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    round-robin policy-name {
        class class {
            match match-name {
                ip {
                    protocol proto
                }
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>proto</i>	Performs a match based on the protocol name (for example, <b>icmp</b> ) or number, as assigned by the IANA.

---

### Default

If not set, packets are not matched against IP protocol.

---

### Usage Guidelines

Use this command to define a match condition for a traffic class based on protocol.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on protocol.

Use the **delete** form of this command to remove protocol value as a match condition.

Use the **show** form of this command to match condition protocol configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ip source

Specifies a match criterion based on source IP information.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ip source
{address ipv4net | port port}

delete traffic-policy round-robin policy-name class class match match-name ip
source {address | port}

show traffic-policy round-robin policy-name class class match match-name ip source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        ip {
          source {
            address ipv4net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>ipv4net</i>	The source IP subnet address to match for this rule.

---

<i>port</i>	The source port to match for this rule. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.
-------------	---

---

---

### Default

If not set, packets are not matched against IP source information.

---

### Usage Guidelines

Use this command to define a match condition based on source subnet address and/or port for a traffic class.

You can match packets based on a source represented by either or both of IP subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 destination

Specifies a match criterion based on IPv6 destination information.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ipv6
destination {address ipv6net | port port}

delete traffic-policy round-robin policy-name class class match match-name ipv6
destination [address | port]

show traffic-policy round-robin policy-name class class match match-name ipv6
destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        ipv6 {
          destination {
            address ipv6net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.



---

<i>ipv6net</i>	Performs a match based on the destination IPv6 subnet address.
<i>port</i>	Performs a match based on destination port. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.

---

---

### Default

If not set, packets are not matched against destination information.

---

### Usage Guidelines

Use this command to define a match condition based on destination subnet address and/or port for a traffic class.

You can match packets based on a destination represented by either or both of IPv6 subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## **traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 dscp <value>**

Specifies a match criterion based on the value of the DSCP field.

---

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ipv6 dscp  
value
```

```
delete traffic-policy round-robin policy-name class class match match-name ipv6  
dscp
```

```
show traffic-policy round-robin policy-name class class match match-name ipv6  
dscp
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    round-robin policy-name {  
        class class {  
            match match-name {  
                ipv6 {  
                    dscp value  
                }  
            }  
        }  
    }  
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.

---

---

<i>value</i>	Performs a match based on the specified value. This value is compared with the value in the DSCP field of the ToS byte in the IP header. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <code>/etc/iproute2/route_dsfld</code> (for example, <code>lowdelay</code> ).
--------------	---

---

---

### Default

If not set, packets are not matched against DSCP value.

---

### Usage Guidelines

Use this command to define a match condition based on the Differentiated Services Code Point (DSCP) field.

The DSCP field is a 6-bit field in the Type of Service (ToS) byte of the IP header. It provides a way of marking packets in order to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on DSCP value.

Use the **delete** form of this command to remove DSCP as a match condition.

Use the **show** form of this command to display DSCP value configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 protocol <proto>

Specifies a match criterion based on the IPv6 protocol.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ipv6
protocol proto
delete traffic-policy round-robin policy-name class class match match-name ipv6
protocol
show traffic-policy round-robin policy-name class class match match-name ipv6
protocol
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        ipv6 {
          protocol proto
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>proto</i>	Performs a match based on the protocol name (for example, <b>icmp</b> ) or number, as assigned by the IANA.

---

### Default

If not set, packets are not matched against IP protocol.

---

### Usage Guidelines

Use this command to define a match condition for a traffic class based on protocol.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on protocol.

Use the **delete** form of this command to remove protocol value as a match condition.

Use the **show** form of this command to match condition protocol configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> ipv6 source

Specifies a match criterion based on source IPv6 information.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name ipv6 source
{address ipv6net | port port}
```

```
delete traffic-policy round-robin policy-name class class match match-name ipv6
source {address | port}
```

```
show traffic-policy round-robin policy-name class class match match-name ipv6
source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        ipv6 {
          source {
            address ipv6net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 3 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.

---

<i>ipv6net</i>	The source IPv6 subnet address to match for this rule.
<i>port</i>	The source port to match for this rule. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.

---

---

### Default

If not set, packets are not matched against IPv6 source information.

---

### Usage Guidelines

Use this command to define a match condition based on source subnet address and/or port for a traffic class.

You can match packets based on a source represented by either or both of IPv6 subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> mark <fwmark>

Specifies a match criterion based on the value of the fwmark field.

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name mark
fwmark
```

```
delete traffic-policy round-robin policy-name class class match match-name mark
```

```
show traffic-policy round-robin policy-name class class match match-name mark
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    class class {
      match match-name {
        mark fwmark
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>fwmark</i>	Performs a match based on the fwmark field. The range is 0 to 4294967295.

### Default

If not set, packets are not matched against fwmark.



---

### Usage Guidelines

Use this command to define a match condition based on the fwmark field for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a fwmark value to be matched.

Use the **delete** form of this command to remove mark as a match condition.

Use the **show** form of this command to display the mark match condition configuration.

## traffic-policy round-robin <policy-name> class <class> match <match-name> vif <vlan-id>

Specifies a match criterion based on VLAN ID.

---

### Syntax

```
set traffic-policy round-robin policy-name class class match match-name vif vlan-id  
delete traffic-policy round-robin policy-name class class match match-name vif  
show traffic-policy round-robin policy-name class class match match-name vif
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    round-robin policy-name {  
        class class {  
            match match-name {  
                vif vlan-id  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>vlan-id</i>	Performs a match based on VLAN ID. The range is 1 to 4096.

---

### Default

If not set, packets are not matched against VLAN ID.

---

### Usage Guidelines

Use this command to define a match condition based on VLAN ID for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a VLAN ID to be matched.

Use the **delete** form of this command to remove VLAN ID as a match condition.

Use the **show** form of this command to display VLAN ID match condition configuration.

## traffic-policy round-robin <policy-name> class <class> quantum <packets>

Specifies the number of packets that can be sent per scheduling quantum for a traffic class.

---

### Syntax

```
set traffic-policy round-robin policy-name class class quantum packets  
delete traffic-policy round-robin policy-name class class quantum  
show traffic-policy round-robin policy-name class class quantum
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    round-robin policy-name {  
        class class {  
            quantum packets  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>packets</i>	Optional. The number of packets that can be sent per scheduling quantum.

---

### Default

None.

---

### Usage Guidelines

Use this command to set the number of packets that can be sent per scheduling quantum for a round robin QoS traffic class policy.

Use the **set** form of this command to specify the number of packets that can be sent per scheduling quantum.

Use the **delete** form of this command to remove the quantum configuration.

Use the **show** form of this command to display the quantum configuration.

## traffic-policy round-robin <policy-name> class <class> queue-limit <limit>

Specifies the maximum queue size for a traffic class.

---

### Syntax

```
set traffic-policy round-robin policy-name class class queue-limit limit  
delete traffic-policy round-robin policy-name class class queue-limit  
show traffic-policy round-robin policy-name class class queue-limit
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    round-robin policy-name {  
        class class {  
            queue-limit limit  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>limit</i>	The maximum queue size in packets.

---

### Default

The default limit is 127.

---

### Usage Guidelines

Use this command to set the maximum queue size (in packets) for a traffic class.  
Use the **set** form of this command to specify the queue limit.

Use the **delete** form of this command to remove queue limit.

Use the **show** form of this command to display queue limit configuration.

## traffic-policy round-robin <policy-name> class <class> queue-type <type>

Specifies the type of queuing to use for a traffic class.

---

### Syntax

```
set traffic-policy round-robin policy-name class class queue-type type  
delete traffic-policy round-robin policy-name class class queue-type  
show traffic-policy round-robin policy-name class class queue-type
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    round-robin policy-name {  
        class class {  
            queue-type type  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>type</i>	The queuing method to use. Supported values are as follows: <b>fair-queue:</b> Uses a Stochastic Fair Queue (SFQ) queue. <b>drop-tail:</b> Uses a First In First Out (FIFO) queue. <b>priority:</b> Sets queue priority based on the Differentiated Services Code Point (DSCP) values in the Type of Service (ToS) byte of the IP header.

---

### Default

The default is **drop-tail**.



---

### Usage Guidelines

Use this command to set the type of queuing mechanism to use for a traffic class.

Use the **set** form of this command to specify the queue type.

Use the **delete** form of this command to restore the default queue type.

Use the **show** form of this command to display queue type configuration.

## traffic-policy round-robin <policy-name> default

Defines a default round robin QoS policy.

---

### Syntax

```
set traffic-policy round-robin policy-name default
delete traffic-policy round-robin policy-name default
show traffic-policy round-robin policy-name default
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    round-robin policy-name {
        default {
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
--------------------	--

---

### Default

None.

---

### Usage Guidelines

Use this command to define a default round robin policy. This policy will be applied to all traffic that does not match any other defined class.

Use the **set** form of this command to create the default class configuration node.

Use the **delete** form of this command to remove the default class configuration node.

Use the **show** form of this command to display the default class configuration node.

## traffic-policy round-robin <policy-name> default quantum <packets>

Specifies the number of packets that can be sent per scheduling quantum.

---

### Syntax

```
set traffic-policy round-robin policy-name default quantum packets
delete traffic-policy round-robin policy-name default quantum
show traffic-policy round-robin policy-name default quantum
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    round-robin policy-name {
        default {
            quantum packets
        }
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>packets</i>	Optional. The number of packets that can be sent per scheduling quantum.

---

---

### Default

None.

---

### Usage Guidelines

Use this command to set the number of packets that can be sent per scheduling quantum for a round robin QoS default policy.

Use the **set** form of this command to specify the number of packets that can be sent per scheduling quantum.

Use the **delete** form of this command to remove the quantum configuration.

Use the **show** form of this command to display the quantum configuration.

## traffic-policy round-robin <policy-name> default queue-limit <limit>

Specifies the maximum queue size for the default traffic class.

---

### Syntax

```
set traffic-policy round-robin policy-name default queue-limit limit
delete traffic-policy round-robin policy-name default queue-limit
show traffic-policy round-robin policy-name default queue-limit
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    default {
      queue-limit limit
    }
  }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>limit</i>	The maximum queue size in packets.

---

### Default

None.

---

### Usage Guidelines

Use this command to set the maximum queue size (in packets) for the default class.

Use the **set** form of this command to specify the queue limit.

Use the **delete** form of this command to remove queue limit.

Use the **show** form of this command to display queue limit configuration.

## traffic-policy round-robin <policy-name> default queue-type <type>

Specifies the type of queuing to use for the default traffic class.

---

### Syntax

```
set traffic-policy round-robin policy-name default queue-type type
delete traffic-policy round-robin policy-name default queue-type
show traffic-policy round-robin policy-name default queue-type
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  round-robin policy-name {
    default {
      queue-type type
    }
  }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>type</i>	The queuing method to use. Supported values are as follows: <b>fair-queue:</b> Uses a Stochastic Fair Queue (SFQ) queue. <b>drop-tail:</b> Uses a First In First Out (FIFO) queue. <b>priority:</b> Sets queue priority based on the Differentiated Services Code Point (DSCP) values in the Type of Service (ToS) byte of the IP header.

---

### Default

The default is **fair-queue**.

---

### Usage Guidelines

Use this command to set the type of queuing mechanism to use for the default traffic class.

Use the **set** form of this command to specify the queue type.

Use the **delete** form of this command to restore the default queue type.

Use the **show** form of this command to display queue type configuration.



## traffic-policy round-robin <policy-name> description <desc>

Specifies a description for a round robin QoS policy.

---

### Syntax

```
set traffic-policy round-robin policy-name description desc  
delete traffic-policy round-robin policy-name description  
show traffic-policy round-robin policy-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    round-robin policy-name {  
        description desc  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the round robin policy.
<i>desc</i>	The description for this round robin policy.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a round robin policy.

Use the **set** form of this command to specify a description for a round robin policy.

Use the **delete** form of this command to remove a description from a round robin policy.

Use the **show** form of this command to display description configuration for a round robin policy.

## traffic-policy shaper <policy-name>

Defines a traffic shaping QoS policy.

---

### Syntax

```
set traffic-policy shaper policy-name
delete traffic-policy shaper policy-name
show traffic-policy shaper policy-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
--------------------	--

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a traffic shaper QoS policy. Traffic shaper policy acts on outbound traffic only. The policy name must be unique and not used with other QoS policy commands.

The Vyatta system uses a version of the Token Bucket traffic shaping algorithm. The Token Bucket algorithm places a limit on the average traffic transmission rate, but allows controlled bursting on the network. The Token Bucket algorithm provides the ability to control bandwidth for VoIP, or limit bandwidth consumption for peer-to-peer applications.

In the Token Bucket algorithm, each flow has a certain number of tokens in its “bucket,” and transmitting traffic “spends” these tokens. If the token bucket is empty, the flow is not permitted to send packets.

This method allows a network administrator to control the amount of bandwidth allocated to different types of traffic. This method also allows a flow to burst traffic, provided it has enough tokens in its bucket.

The difference between **shaper** and **round-robin** is that **shaper** limits bandwidth usage by class and then allocates any leftover bandwidth. **Round-robin**, on the other hand, attempts to divide all available bandwidth between the defined classes.

Use the **set** form of this command to create a traffic shaper QoS policy.

Use the **delete** form of this command to remove a traffic shaper QoS policy.

Use the **show** form of this command to display traffic shaper QoS policy configuration.

## traffic-policy shaper <policy-name> bandwidth

Specifies the bandwidth available for all combined traffic constrained by this policy.

---

### Syntax

```
set traffic-policy shaper policy-name bandwidth [auto | rate | rate-suffix]
delete traffic-policy shaper policy-name bandwidth
show traffic-policy shaper policy-name bandwidth
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        bandwidth [auto | rate | rate-suffix]
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>auto</i>	Automatically bases the bandwidth on the interface speed.
<i>rate</i>	The bandwidth, specified in kilobits per second.
<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported: <b>kbit</b> : Kilobits per second. <b>mbit</b> : Megabits per second. <b>gbit</b> : Gigabits per second. <b>kbps</b> : Kilobytes per second. <b>mbps</b> : Megabytes per second. <b>gbps</b> : Gigabytes per second.

---

---

### Default

The default is **auto**.

---

### Usage Guidelines

Use this command to set bandwidth constraints for a traffic shaper QoS policy. This is the maximum bandwidth available for all classes.

Use the **set** form of this command to specify bandwidth constraints for the policy.

Use the **delete** form of this command to restore default bandwidth constraints for the policy.

Use the **show** form of this command to display policy bandwidth configuration.

## traffic-policy shaper <policy-name> class <class>

Defines a traffic class for a traffic shaper QoS policy.

---

### Syntax

```
set traffic-policy shaper policy-name class class
delete traffic-policy shaper policy-name class class
show traffic-policy shaper policy-name class class
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        class class {
        }
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.

---

---

### Default

None.

---

### Usage Guidelines

Use this command to define a traffic class for a traffic shaper QoS policy. This allows packets to be grouped into various traffic classes, which can be treated with different levels of service.

Use the set form of this command to create a traffic class in a traffic shaper QoS policy.

Use the **delete** form of this command to remove a traffic class from a traffic shaper QoS policy.

Use the **show** form of this command to display traffic class configuration within a traffic shaper QoS policy.



## traffic-policy shaper <policy-name> class <class> bandwidth

Specifies the base guaranteed bandwidth rate for a traffic class.

---

### Syntax

```
set traffic-policy shaper policy-name class class bandwidth [rate | rate-pct |  
rate-suffix]
```

```
delete traffic-policy shaper policy-name class class bandwidth
```

```
show traffic-policy shaper policy-name class class bandwidth
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        class class {  
            bandwidth [rate | rate-pct | rate-suffix]  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>rate</i>	The bandwidth, specified in kilobits per second.
<i>rate-pct</i>	The bandwidth, specified as a percentage of the overall bandwidth rate. The format is <i>num</i> % (for example, 85%).

---

<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported: <b>kbit</b> : Kilobits per second. <b>mbit</b> : Megabits per second. <b>gbit</b> : Gigabits per second. <b>kbps</b> : Kilobytes per second. <b>mbps</b> : Megabytes per second. <b>gbps</b> : Gigabytes per second.
--------------------	---

---

---

### Default

100% bandwidth usage is available.

---

### Usage Guidelines

Use this command to set a base level of guaranteed bandwidth for a traffic class.

Use the **set** form of this command to set the available bandwidth for the traffic class.

Use the **delete** form of this command to restore the default available bandwidth for the traffic class.

Use the **show** form of this command to display class bandwidth configuration.

## traffic-policy shaper <policy-name> class <class> burst

Sets the burst size for a traffic class.

---

### Syntax

```
set traffic-policy shaper policy-name class class burst [num | num-suffix]  
delete traffic-policy shaper policy-name class class burst  
show traffic-policy shaper policy-name class class burst
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        class class {  
            burst [num | num-suffix]  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>num</i>	The burst size, specified in bytes.
<i>num-suffix</i>	The burst size, specified as a number and a scaling suffix (for example, 10mb). The following suffixes are supported: <b>kb</b> : Kilobytes. <b>mb</b> : Megabytes. <b>gb</b> : Gigabytes.

---

### Default

The burst size is 15 kilobytes.

---

### Usage Guidelines

Use this command to set the burst size for the traffic class. This is the maximum amount of traffic that may be sent at a given time.

Use the **set** form of this command to specify the burst size for a traffic class.

Use the **delete** form of this command to restore the default burst size for a traffic class.

Use the **show** form of this command to display traffic class burst size configuration.

## traffic-policy shaper <policy-name> class <class> ceiling

Sets a bandwidth ceiling for a traffic class.

### Syntax

```
set traffic-policy shaper policy-name class class ceiling [rate | rate-pct | rate-suffix]
delete traffic-policy shaper policy-name class class ceiling
show traffic-policy shaper policy-name class class ceiling
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        class class {
            ceiling [rate | rate-pct | rate-suffix]
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>rate</i>	The maximum bandwidth, specified in kilobits per second.
<i>rate-pct</i>	The maximum bandwidth, specified as a percentage of the interface speed. The format is <i>num</i> % (for example, 85%).
<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported: <b>kbit</b> : Kilobits per second. <b>mbit</b> : Megabits per second. <b>gbit</b> : Gigabits per second.

---

### Default

The default is the specified bandwidth for the class.

---

### Usage Guidelines

Use this command to set the maximum amount of bandwidth a traffic class may consume when excess bandwidth is available.

Use the **set** form of this command to set the bandwidth ceiling for a traffic class.

Use the **delete** form of this command to restore the default bandwidth ceiling for a traffic class.

Use the **show** form of this command to display traffic class bandwidth ceiling configuration.

## traffic-policy shaper <policy-name> class <class> description <desc>

Sets a description for a traffic class.

---

### Syntax

```
set traffic-policy shaper policy-name class class description desc  
delete traffic-policy shaper policy-name class class description  
show traffic-policy shaper policy-name class class description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        class class {  
            description desc  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>desc</i>	The description for this traffic class.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic class.

Use the **set** form of this command to specify a description.

Use the **delete** form of this command to remove a description.

Use the **show** form of this command to display description configuration.



## traffic-policy shaper <policy-name> class <class> match <match-name>

Defines a traffic class matching rule.

---

### Syntax

```
set traffic-policy shaper policy-name class class match match-name
delete traffic-policy shaper policy-name class class match match-name
show traffic-policy shaper policy-name class class match match-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    class class {
      match match-name {
      }
    }
  }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.

---

### Default

None.

---

### Usage Guidelines

Use this command to define a rule setting out the match conditions for membership in a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to create the traffic class matching rule. Note that you cannot use **set** to change the name of an existing traffic class matching rule. To change the rule, delete it and re-create it.

Use the **delete** form of this command to remove the traffic class matching rule configuration node.

Use the **show** form of this command to display traffic class matching rule configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> description <desc>

Sets a description for a match rule.

---

### Syntax

```
set traffic-policy shaper policy-name class class match match-name description desc  
delete traffic-policy shaper policy-name class class match match-name description  
show traffic-policy shaper policy-name class class match match-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        class class {  
            match match-name {  
                description desc  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>desc</i>	The description for this match.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic class matching rule.

Use the **set** form of this command to set the description.

Use the **delete** form of this command to remove the description.

Use the **show** form of this command to display description configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> ether destination <mac-addr>

Specifies a match criterion based on Ethernet destination (MAC) address.

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ether destination mac-addr
```

```
delete traffic-policy shaper policy-name class class match match-name ether destination
```

```
show traffic-policy shaper policy-name class class match match-name ether destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        class class {
            match match-name {
                ether {
                    destination mac-addr
                }
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>mac-addr</i>	Performs a match based on the destination MAC address on the interface the policy is applied to. The format is 6 colon-separated 8-bit numbers in hexadecimal; for example, 00:0a:59:9a:f2:ba.

---

### Default

If not set, packets are not matched against the destination MAC address.

---

### Usage Guidelines

Use this command to define a match condition based on destination MAC address for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> ether protocol <num>

Specifies a match criterion based on Ethernet packet type.

---

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ether protocol num
```

```
delete traffic-policy shaper policy-name class class match match-name ether protocol
```

```
show traffic-policy shaper policy-name class class match match-name ether protocol
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        class class {  
            match match-name {  
                ether {  
                    protocol num  
                }  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>num</i>	Performs a match based on the Ethernet packet type (i.e. protocol number). The range is 0 to 65535.

---

### Default

If not set, packets are not matched against the Ethernet packet type.

---

### Usage Guidelines

Use this command to define a match condition based on Ethernet packet type for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify the packet type to be matched.

Use the **delete** form of this command to remove packet type as a match condition.

Use the **show** form of this command to display packet type match condition configuration.



## traffic-policy shaper <policy-name> class <class> match <match-name> ether source <mac-addr>

Specifies a match criterion based on Ethernet source (MAC) address.

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ether source mac-addr
```

```
delete traffic-policy shaper policy-name class class match match-name ether source
```

```
show traffic-policy shaper policy-name class class match match-name ether source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    class class {
      match match-name {
        ether {
          source mac-addr
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>mac-addr</i>	Performs a match based on the MAC address of the interface the policy is applied to. The format is 6 colon-separated 8-bit numbers in hexadecimal; for example, 00:0a:59:9a:f2:ba.

---

### Default

If not set, packets are not matched against the source MAC address.

---

### Usage Guidelines

Use this command to define a match condition based on source MAC address for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> interface <interface>

Specifies a match criterion based on incoming interface.

### Syntax

```
set traffic-policy shaper policy-name class class match match-name interface
interface
delete traffic-policy shaper policy-name class class match match-name interface
show traffic-policy shaper policy-name class class match match-name interface
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    class class {
      match match-name {
        interface interface
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>interface</i>	Performs a match based on the specified Ethernet interface name. The ingress interface for incoming traffic will be compared with this value.

---

## Default

None.

---

## Usage Guidelines

Use this command to set a match condition for a traffic class based on incoming interface.

If incoming packets ingress through the interface specified by this command, the traffic is a member of this traffic class (provided other match conditions are satisfied).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify an interface to be matched by incoming packets.

Use the **delete** form of this command to remove the match interface.

Use the **show** form of this command to display interface match configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> ip destination

Specifies a match criterion based on IP destination information.

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ip destination
{address ipv4net | port port}
```

```
delete traffic-policy shaper policy-name class class match match-name ip destination
[address | port]
```

```
show traffic-policy shaper policy-name class class match match-name ip destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    class class {
      match match-name {
        ip {
          destination {
            address ipv4net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>ipv4net</i>	Performs a match based on the destination IP subnet address.

---

<i>port</i>	Performs a match based on destination port. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.
-------------	---

---

---

### Default

If not set, packets are not matched against destination information.

---

### Usage Guidelines

Use this command to define a match condition based on destination subnet address and/or port for a traffic class.

You can match packets based on a destination represented by either or both of IP subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> ip dscp <value>

Specifies a match criterion based on the value of the DSCP field.

---

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ip dscp value
delete traffic-policy shaper policy-name class class match match-name ip dscp
show traffic-policy shaper policy-name class class match match-name ip dscp
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    class class {
      match match-name {
        ip {
          dscp value
        }
      }
    }
  }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>value</i>	Performs a match based on the specified value. This value is compared with the value in the DSCP field of the ToS byte in the IP header. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <i>/etc/iproute2/rt_dsfield</i> (for example, <i>lowdelay</i> ).

---

---

### Default

If not set, packets are not matched against DSCP value.

---

### Usage Guidelines

Use this command to define a match condition based on the Differentiated Services Code Point (DSCP) field.

The DSCP field is a 6-bit field in the Type of Service (ToS) byte of the IP header. It provides a way of marking packets in order to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on DSCP value.

Use the **delete** form of this command to remove DSCP as a match condition.

Use the **show** form of this command to display DSCP value configuration.



## traffic-policy shaper <policy-name> class <class> match <match-name> ip protocol <proto>

Specifies a match criterion based on the IP protocol.

---

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ip protocol proto  
delete traffic-policy shaper policy-name class class match match-name ip protocol  
show traffic-policy shaper policy-name class class match match-name ip protocol
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        class class {  
            match match-name {  
                ip {  
                    protocol proto  
                }  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>proto</i>	Performs a match based on the protocol name (for example, <b>icmp</b> ) or number, as assigned by the IANA.

---

### Default

If not set, packets are not matched against IP protocol.

---

### Usage Guidelines

Use this command to define a match condition for a traffic class based on protocol.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on protocol.

Use the **delete** form of this command to remove protocol value as a match condition.

Use the **show** form of this command to match condition protocol configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> ip source

Specifies a match criterion based on source IP information.

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ip source
{address ipv4net | port port}

delete traffic-policy shaper policy-name class class match match-name ip source
{address | port}

show traffic-policy shaper policy-name class class match match-name ip source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    class class {
      match match-name {
        ip {
          source {
            address ipv4net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>ipv4net</i>	The source IP subnet address to match for this rule.

---

<i>port</i>	The source port to match for this rule. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.
-------------	---

---

---

### Default

If not set, packets are not matched against IP source information.

---

### Usage Guidelines

Use this command to define a match condition based on source subnet address and/or port for a traffic class.

You can match packets based on a source represented by either or both of IP subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 destination

Specifies a match criterion based on IPv6 destination information.

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ipv6 destination
{address ipv6net | port port}
```

```
delete traffic-policy shaper policy-name class class match match-name ipv6
destination [address | port]
```

```
show traffic-policy shaper policy-name class class match match-name ipv6
destination
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    class class {
      match match-name {
        ipv6 {
          destination {
            address ipv6net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.

---

<i>ipv6net</i>	Performs a match based on the destination IPv6 subnet address.
<i>port</i>	Performs a match based on destination port. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.

---

---

### Default

If not set, packets are not matched against destination information.

---

### Usage Guidelines

Use this command to define a match condition based on a destination subnet address and/or port for a traffic class.

You can match packets based on a destination represented by either or both of the IPv6 subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a destination to be matched.

Use the **delete** form of this command to remove destination as a match condition.

Use the **show** form of this command to display destination match condition configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 dscp <value>

Specifies a match criterion based on the value of the DSCP field.

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ipv6 dscp value
delete traffic-policy shaper policy-name class class match match-name ipv6 dscp
show traffic-policy shaper policy-name class class match match-name ipv6 dscp
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        class class {
            match match-name {
                ipv6 {
                    dscp value
                }
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>value</i>	Performs a match based on the specified value. This value is compared with the value in the DSCP field of the ToS byte in the IP header. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <i>/etc/iproute2/rt_dsfield</i> (for example, <i>lowdelay</i> ).

---

### Default

If not set, packets are not matched against DSCP value.

---

### Usage Guidelines

Use this command to define a match condition based on the Differentiated Services Code Point (DSCP) field.

The DSCP field is a 6-bit field in the Type of Service (ToS) byte of the IP header. It provides a way of marking packets in order to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on DSCP value.

Use the **delete** form of this command to remove DSCP as a match condition.

Use the **show** form of this command to display DSCP value configuration.



## traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 protocol <proto>

Specifies a match criterion based on the IPv6 protocol.

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ipv6 protocol
proto
```

```
delete traffic-policy shaper policy-name class class match match-name ipv6 protocol
```

```
show traffic-policy shaper policy-name class class match match-name ipv6 protocol
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        class class {
            match match-name {
                ipv6 {
                    protocol proto
                }
            }
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>proto</i>	Performs a match based on the protocol name (for example, <b>icmp</b> ) or number, as assigned by the IANA.

---

### Default

If not set, packets are not matched against IP protocol.

---

### Usage Guidelines

Use this command to define a match condition for a traffic class based on protocol.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to set a match condition based on protocol.

Use the **delete** form of this command to remove protocol value as a match condition.

Use the **show** form of this command to match condition protocol configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> ipv6 source

Specifies a match criterion based on source IPv6 information.

### Syntax

```
set traffic-policy shaper policy-name class class match match-name ipv6 source
{address ipv6net | port port}

delete traffic-policy shaper policy-name class class match match-name ipv6 source
{address | port}

show traffic-policy shaper policy-name class class match match-name ipv6 source
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    class class {
      match match-name {
        ipv6 {
          source {
            address ipv6net
            port port
          }
        }
      }
    }
  }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>ipv6net</i>	The source IPv6 subnet address to match for this rule.

---

<i>port</i>	The source port to match for this rule. The port may be specified as a lower-case name (for example, <b>ssh</b> ) or as a number. The range for port numbers is 0 to 65535.
-------------	---

---

---

### Default

If not set, packets are not matched against IPv6 source information.

---

### Usage Guidelines

Use this command to define a match condition based on source subnet address and/or port for a traffic class.

You can match packets based on a source represented by either or both of IPv6 subnet address and destination port(s).

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a source to be matched.

Use the **delete** form of this command to remove source as a match condition.

Use the **show** form of this command to display source match condition configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> mark <fwmark>

Specifies a match criterion based on the value of the fwmark field.

---

### Syntax

```
set traffic-policy shaper policy-name class class match match-name mark fwmark  
delete traffic-policy shaper policy-name class class match match-name mark  
show traffic-policy shaper policy-name class class match match-name mark
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        class class {  
            match match-name {  
                mark fwmark  
            }  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>fwmark</i>	Performs a match based on the fwmark field. The range is 0 to 4294967295.

---

### Default

If not set, packets are not matched against fwmark.

---

### Usage Guidelines

Use this command to define a match condition based on the fwmark field for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a fwmark value to be matched.

Use the **delete** form of this command to remove mark as a match condition.

Use the **show** form of this command to display the mark match condition configuration.

## traffic-policy shaper <policy-name> class <class> match <match-name> vif <vlan-id>

Specifies a match criterion based on VLAN ID.

---

### Syntax

```
set traffic-policy shaper policy-name class class match match-name vif vlan-id
delete traffic-policy shaper policy-name class class match match-name vif
show traffic-policy shaper policy-name class class match match-name vif
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    class class {
      match match-name {
        vif vlan-id
      }
    }
  }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>match-name</i>	Mandatory. Class matching rule name.
<i>vlan-id</i>	Performs a match based on VLAN ID. The range is 1 to 4096.

---

### Default

If not set, packets are not matched against VLAN ID.

---

### Usage Guidelines

Use this command to define a match condition based on VLAN ID for a traffic class.

**NOTE** *Interface and vif match rules match on packet meta data. All other match rules match on packet data. Match rules from these two groups cannot be combined.*

Use the **set** form of this command to specify a VLAN ID to be matched.

Use the **delete** form of this command to remove VLAN ID as a match condition.

Use the **show** form of this command to display VLAN ID match condition configuration.



## traffic-policy shaper <policy-name> class <class> priority <priority>

Specifies the priority of a traffic class for allocation of extra bandwidth.

---

### Syntax

```
set traffic-policy shaper policy-name class class priority priority
delete traffic-policy shaper policy-name class class priority
show traffic-policy shaper policy-name class class priority
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        class class {
            priority priority
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>priority</i>	The priority with which this traffic class should be allocated extra bandwidth. The range is 0 to 7, where the lower the number the lower the priority. The default is 0.

---

### Default

Traffic classes are assigned a priority of 0.

---

### Usage Guidelines

Use this command to set the priority with which a traffic class is to be awarded extra bandwidth when excess is available.

Use the **set** form of this command to specify priority for a traffic class.

Use the **delete** form of this command to restore the default priority for a traffic class.

Use the **show** form of this command to display traffic class priority configuration.

## traffic-policy shaper <policy-name> class <class> queue-limit <limit>

Specifies the maximum queue size for a traffic class.

---

### Syntax

```
set traffic-policy shaper policy-name class class queue-limit limit  
delete traffic-policy shaper policy-name class class queue-limit  
show traffic-policy shaper policy-name class class queue-limit
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        class class {  
            queue-limit limit  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>limit</i>	The maximum queue size in packets.

---

### Default

None.

---

### Usage Guidelines

Use this command to set the maximum queue size (in packets) for a traffic class.  
Use the **set** form of this command to specify the queue limit.

Use the **delete** form of this command to remove queue limit.

Use the **show** form of this command to display queue limit configuration.

## traffic-policy shaper <policy-name> class <class> queue-type <type>

Specifies the type of queuing to use for a traffic class.

### Syntax

```
set traffic-policy shaper policy-name class class queue-type type
delete traffic-policy shaper policy-name class class queue-type
show traffic-policy shaper policy-name class class queue-type
```

### Command Mode

Configuration mode.

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        class class {
            queue-type type
        }
    }
}
```

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>type</i>	The queuing method to use. Supported values are as follows: <b>fair-queue:</b> Uses a Stochastic Fair Queue (SFQ) queue. <b>drop-tail:</b> Uses a First In First Out (FIFO) queue. <b>priority:</b> Sets queue priority based on the Differentiated Services Code Point (DSCP) values in the Type of Service (ToS) byte of the IP header. <b>random-detect:</b> Uses a Random Early Detection (RED) queue.

---

### Default

The default is **fair-queue**.

---

### Usage Guidelines

Use this command to set the type of queuing mechanism to use for a traffic class.

Use the **set** form of this command to specify the queue type.

Use the **delete** form of this command to restore the default queue type.

Use the **show** form of this command to display queue type configuration.

## traffic-policy shaper <policy-name> class <class> set-dscp <value>

Rewrites the DSCP field in packets in this traffic class to the specified value.

---

### Syntax

```
set traffic-policy shaper policy-name class class set-dscp value
delete traffic-policy shaper policy-name class class set-dscp
show traffic-policy shaper policy-name class class set-dscp
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        class class {
            set-dscp value
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>class</i>	Mandatory. The class ID. The range is 2 to 4095.
<i>value</i>	The value to write into the DSCP field of packets in this traffic class. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <code>/etc/iproute2/rt_dsfield</code> (for example, <code>lowdelay</code> ). By default, the DSCP field is not rewritten.

---

### Default

If not set, the DSCP byte is not rewritten.

## Usage Guidelines

Use this command to direct the system to rewrite the Differentiated Services Code Point (DSCP) field of packets in a traffic class to a specific value.

Rewriting the DSCP field can be a way to specify forwarding behavior of a network for packets to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

The following table shows the standard semantics for DSCP values, as specified by RFC 2474.

Table 3-1 RFC 2474 DSCP Values

Binary Value	Configured Value	Drop Rate	Meaning
101 110	46	N/A	Expedited forwarding (EF)
000 000	0	N/A	Default: Best-effort traffic
001 010	10	Low	Assured forwarding (AF) 11
001 100	12	Medium	Assured forwarding (AF) 12
001 110	14	High	Assured forwarding (AF) 13
010 010	18	Low	Assured forwarding (AF) 21
010 100	20	Medium	Assured forwarding (AF) 22
010 110	22	High	Assured forwarding (AF) 23
011 010	26	Low	Assured forwarding (AF) 31
011 100	28	Medium	Assured forwarding (AF) 32
011 110	30	High	Assured forwarding (AF) 33
100 010	34	Low	Assured forwarding (AF) 41
100 100	36	Medium	Assured forwarding (AF) 42
100 110	38	High	Assured forwarding (AF) 43

Use the **set** form of this command to rewrite DSCP values of packets in a traffic class.

Use the **delete** form of this command to stop DSCP values from being rewritten.

Use the **show** form of this command to display DSCP rewrite configuration.



## traffic-policy shaper <policy-name> default

Defines a default traffic shaper QoS policy.

---

### Syntax

```
set traffic-policy shaper policy-name default
delete traffic-policy shaper policy-name default
show traffic-policy shaper policy-name default
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        default {
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
--------------------	--

---

### Default

None.

---

### Usage Guidelines

Use this command to define a default traffic shaping policy. This policy will be applied to all traffic that does not match any other defined class.

Use the **set** form of this command to create the default class configuration node.

Use the **delete** form of this command to remove the default class configuration node.

Use the **show** form of this command to display the default class configuration node.

## traffic-policy shaper <policy-name> default bandwidth

Specifies the base guaranteed bandwidth rate for the default traffic class.

---

### Syntax

```
set traffic-policy shaper policy-name default bandwidth [rate | rate-pct | rate-suffix]
delete traffic-policy shaper policy-name default bandwidth
show traffic-policy shaper policy-name default bandwidth
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        default {
            bandwidth [rate | rate-pct | rate-suffix]
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>rate</i>	Bandwidth in kbps.
<i>rate-pct</i>	The bandwidth, specified as a percentage of the interface speed. The format is <i>num</i> % (for example, 85%).
<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported: <b>kbit</b> : Kilobits per second. <b>mbit</b> : Megabits per second. <b>gbit</b> : Gigabits per second. <b>kbps</b> : Kilobytes per second. <b>mbps</b> : Megabytes per second. <b>gbps</b> : Gigabytes per second.

---

---

### Default

100% bandwidth usage is available.

---

### Usage Guidelines

Use this command to set a base level of guaranteed bandwidth for the default traffic class.

Use the **set** form of this command to set the available bandwidth for the default traffic class.

Use the **delete** form of this command to restore the default available bandwidth for the default traffic class.

Use the **show** form of this command to display bandwidth configuration for the default traffic class.

## traffic-policy shaper <policy-name> default burst

Sets the burst size for the default traffic class.

---

### Syntax

```
set traffic-policy shaper policy-name default burst [num | num-suffix]  
delete traffic-policy shaper policy-name default burst  
show traffic-policy shaper policy-name default burst
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        default {  
            burst [num | num-suffix]  
        }  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>num</i>	Burst size in bytes.
<i>num-suffix</i>	The burst size, specified as a number and a scaling suffix (for example, 10mb). The following suffixes are supported: <b>kb</b> : Kilobytes. <b>mb</b> : Megabytes. <b>gb</b> : Gigabytes.

---

### Default

The burst size is 15kb.

---

### Usage Guidelines

Use this command to set the burst size for the default traffic class. This is the maximum amount of traffic that may be sent at a given time.

Use the **set** form of this command to specify the burst size for the default traffic class.

Use the **delete** form of this command to restore the default burst size for the default traffic class.

Use the **show** form of this command to display burst size configuration for the default traffic class.

## traffic-policy shaper <policy-name> default ceiling

Sets a bandwidth ceiling for the default traffic class.

---

### Syntax

```
set traffic-policy shaper policy-name default ceiling [rate | rate-pct | rate-suffix]
delete traffic-policy shaper policy-name default ceiling
show traffic-policy shaper policy-name default ceiling
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    default {
      ceiling [rate | rate-pct | rate-suffix]
    }
  }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>rate</i>	The limit in kbps.
<i>rate-pct</i>	The bandwidth, specified as a percentage of the overall bandwidth rate. The format is <i>num</i> % (for example, 85%).
<i>rate-suffix</i>	The bandwidth, specified as a number and a scaling suffix (for example, 10mbit). The following suffixes are supported: <b>kbit</b> : Kilobits per second. <b>mbit</b> : Megabits per second. <b>gbit</b> : Gigabits per second.

---

### Default

The default is the full bandwidth.

---

### Usage Guidelines

Use this command to set the maximum amount of bandwidth the default traffic class may consume when excess bandwidth is available.

Use the **set** form of this command to set the bandwidth ceiling for the default traffic class.

Use the **delete** form of this command to restore the default bandwidth ceiling for the default traffic class.

Use the **show** form of this command to display bandwidth ceiling configuration for the default traffic class.

## traffic-policy shaper <policy-name> default priority <priority>

Specifies the priority of the default traffic class for allocation of extra bandwidth.

---

### Syntax

```
set traffic-policy shaper policy-name default priority priority
delete traffic-policy shaper policy-name default priority
show traffic-policy shaper policy-name default priority
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    default {
      priority priority
    }
  }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>priority</i>	The priority with which this traffic class should be allocated extra bandwidth. The range is 0 to 7, where the higher the number the lower the priority. The default is 0.

---

### Default

The default priority is assigned a value of 0.

---

### Usage Guidelines

Use this command to set the priority with which the default traffic class is to be awarded extra bandwidth when excess is available.



Use the **set** form of this command to specify priority for the default traffic class.

Use the **delete** form of this command to restore the default priority for the default traffic class.

Use the **show** form of this command to display priority configuration for the default traffic class.

## traffic-policy shaper <policy-name> default queue-limit <limit>

Specifies the maximum queue size for the default traffic class.

---

### Syntax

```
set traffic-policy shaper policy-name default queue-limit limit
delete traffic-policy shaper policy-name default queue-limit
show traffic-policy shaper policy-name default queue-limit
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
  shaper policy-name {
    default {
      queue-limit limit
    }
  }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>limit</i>	The maximum queue size in packets.

---

---

### Default

None.

---

### Usage Guidelines

Use this command to set the maximum queue size (in packets) for the default class.

Use the **set** form of this command to specify the queue limit.

Use the **delete** form of this command to remove queue limit.

Use the **show** form of this command to display queue limit configuration.

## traffic-policy shaper <policy-name> default queue-type <type>

Specifies the type of queuing to use for the default traffic class.

---

### Syntax

```
set traffic-policy shaper policy-name default queue-type type
delete traffic-policy shaper policy-name default queue-type
show traffic-policy shaper policy-name default queue-type
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        default {
            queue-type type
        }
    }
}
```

---

### Parameters

---

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>type</i>	The queuing method to use. Supported values are as follows: <b>fair-queue:</b> Uses a Stochastic Fair Queue (SFQ) queue. <b>drop-tail:</b> Uses a First In First Out (FIFO) queue. <b>priority:</b> Sets queue priority based on the Differentiated Services Code Point (DSCP) values in the Type of Service (ToS) byte of the IP header. <b>random-detect:</b> Uses a Random Early Detection (RED) queue.

---

---

### Default

The default is **fair-queue**.

---

### Usage Guidelines

Use this command to set the type of queuing mechanism to use for the default traffic class.

Use the **set** form of this command to specify the queue type.

Use the **delete** form of this command to restore the default queue type.

Use the **show** form of this command to display queue type configuration.

## traffic-policy shaper <policy-name> default set-dscp <value>

Rewrites the DSCP field in packets in the default traffic class to the specified value.

---

### Syntax

```
set traffic-policy shaper policy-name default set-dscp value
delete traffic-policy shaper policy-name default set-dscp
show traffic-policy shaper policy-name default set-dscp
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {
    shaper policy-name {
        default {
            set-dscp value
        }
    }
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>value</i>	The value to write into the DSCP field of packets in the default traffic class. The DSCP value can be specified as a decimal number (for example, 12), as a hexadecimal number (for example, 0x1D), or as a standard name from <i>/etc/iproute2/rt_dsfield</i> (for example, <i>lowdelay</i> ). By default, the DSCP field is not rewritten.

---

### Default

If not set, the DSCP byte is not rewritten.

## Usage Guidelines

Use this command to direct the system to rewrite the Differentiated Services Code Point (DSCP) field of packets in the default traffic class to a specific value.

Rewriting the DSCP field can be a way to specify forwarding behavior of a network for packets to allow classification of traffic into service classes, and traffic conditioning such as metering, policing, and shaping.

The following table shows the standard semantics for DSCP values, as specified by RFC 2474.

Table 3-2 RFC 2474 DSCP Values

Binary Value	Configured Value	Drop Rate	Meaning
101 110	46	N/A	Expedited forwarding (EF)
000 000	0	N/A	Default: Best-effort traffic
001 010	10	Low	Assured forwarding (AF) 11
001 100	12	Medium	Assured forwarding (AF) 12
001 110	14	High	Assured forwarding (AF) 13
010 010	18	Low	Assured forwarding (AF) 21
010 100	20	Medium	Assured forwarding (AF) 22
010 110	22	High	Assured forwarding (AF) 23
011 010	26	Low	Assured forwarding (AF) 31
011 100	28	Medium	Assured forwarding (AF) 32
011 110	30	High	Assured forwarding (AF) 33
100 010	34	Low	Assured forwarding (AF) 41
100 100	36	Medium	Assured forwarding (AF) 42
100 110	38	High	Assured forwarding (AF) 43

Use the **set** form of this command to rewrite DSCP values of packets in the default traffic class.

Use the **delete** form of this command to stop DSCP values in the default traffic class from being rewritten.

Use the **show** form of this command to display DSCP rewrite configuration for the default traffic class.

## traffic-policy shaper <policy-name> description <desc>

Specifies a description for a traffic shaper QoS policy.

---

### Syntax

```
set traffic-policy shaper policy-name description desc  
delete traffic-policy shaper policy-name description  
show traffic-policy shaper policy-name description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
traffic-policy {  
    shaper policy-name {  
        description desc  
    }  
}
```

---

### Parameters

<i>policy-name</i>	Mandatory. The name of the traffic shaping policy.
<i>desc</i>	The description for this traffic shaper policy.

---

### Default

None.

---

### Usage Guidelines

Use this command to record a description for a traffic shaper policy.

Use the **set** form of this command to specify a description for a traffic shaper policy.

Use the **delete** form of this command to remove a description from a traffic shaper policy.

Use the **show** form of this command to display description configuration for a traffic shaper policy.



---

## Glossary of Acronyms

ACL	access control list
ADSL	Asymmetric Digital Subscriber Line
AH	Authentication Header
AMI	Amazon Machine Image
API	Application Programming Interface
AS	autonomous system
ARP	Address Resolution Protocol
AWS	Amazon Web Services
BGP	Border Gateway Protocol
BIOS	Basic Input Output System
BPDU	Bridge Protocol Data Unit
CA	certificate authority
CCMP	AES in counter mode with CBC-MAC
CHAP	Challenge Handshake Authentication Protocol
CLI	command-line interface
DDNS	dynamic DNS
DHCP	Dynamic Host Configuration Protocol

---

DHCPv6	Dynamic Host Configuration Protocol version 6
DLCI	data-link connection identifier
DMI	desktop management interface
DMVPN	dynamic multipoint VPN
DMZ	demilitarized zone
DN	distinguished name
DNS	Domain Name System
DSCP	Differentiated Services Code Point
DSL	Digital Subscriber Line
eBGP	external BGP
EBS	Amazon Elastic Block Storage
EC2	Amazon Elastic Compute Cloud
EGP	Exterior Gateway Protocol
ECMP	equal-cost multipath
ESP	Encapsulating Security Payload
FIB	Forwarding Information Base
FTP	File Transfer Protocol
GRE	Generic Routing Encapsulation
HDLC	High-Level Data Link Control
I/O	Input/Output
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IGMP	Internet Group Management Protocol
IGP	Interior Gateway Protocol
IPS	Intrusion Protection System

---

IKE	Internet Key Exchange
IP	Internet Protocol
IPOA	IP over ATM
IPsec	IP Security
IPv4	IP Version 4
IPv6	IP Version 6
IS-IS	Intermediate System-to-Intermediate System
ISAKMP	Internet Security Association and Key Management Protocol
ISM	Internet Standard Multicast
ISP	Internet Service Provider
KVM	Kernel-Based Virtual Machine
L2TP	Layer 2 Tunneling Protocol
LACP	Link Aggregation Control Protocol
LAN	local area network
LDAP	Lightweight Directory Access Protocol
LLDP	Link Layer Discovery Protocol
MAC	medium access control
mGRE	multipoint GRE
MIB	Management Information Base
MLD	Multicast Listener Discovery
MLPPP	multilink PPP
MPLS	Multiprotocol Label Switching
MPLS EXP	MPLS experimental
MPLS TE	MPLS Traffic Engineering
MRRU	maximum received reconstructed unit
MTU	maximum transmission unit

---

NAT	Network Address Translation
NBMA	Non-Broadcast Multi-Access
ND	Neighbor Discovery
NHRP	Next Hop Resolution Protocol
NIC	network interface card
NTP	Network Time Protocol
OSPF	Open Shortest Path First
OSPFv2	OSPF Version 2
OSPFv3	OSPF Version 3
PAM	Pluggable Authentication Module
PAP	Password Authentication Protocol
PAT	Port Address Translation
PCI	peripheral component interconnect
PIM	Protocol Independent Multicast
PIM-DM	PIM Dense Mode
PIM-SM	PIM Sparse Mode
PKI	Public Key Infrastructure
PPP	Point-to-Point Protocol
PPPoA	PPP over ATM
PPPoE	PPP over Ethernet
PPTP	Point-to-Point Tunneling Protocol
PTMU	Path Maximum Transfer Unit
PVC	permanent virtual circuit
QoS	quality of service
RADIUS	Remote Authentication Dial-In User Service
RHEL	Red Hat Enterprise Linux

---

---

RIB	Routing Information Base
RIP	Routing Information Protocol
RIPng	RIP next generation
RP	Rendezvous Point
RPF	Reverse Path Forwarding
RSA	Rivest, Shamir, and Adleman
Rx	receive
S3	Amazon Simple Storage Service
SLAAC	Stateless Address Auto-Configuration
SNMP	Simple Network Management Protocol
SMTP	Simple Mail Transfer Protocol
SONET	Synchronous Optical Network
SPT	Shortest Path Tree
SSH	Secure Shell
SSID	Service Set Identifier
SSM	Source-Specific Multicast
STP	Spanning Tree Protocol
TACACS+	Terminal Access Controller Access Control System Plus
TBF	Token Bucket Filter
TCP	Transmission Control Protocol
TKIP	Temporal Key Integrity Protocol
ToS	Type of Service
TSS	TCP Maximum Segment Size
Tx	transmit
UDP	User Datagram Protocol
VHD	virtual hard disk

---

---

vif	virtual interface
VLAN	virtual LAN
VPC	Amazon virtual private cloud
VPN	virtual private network
VRRP	Virtual Router Redundancy Protocol
WAN	wide area network
WAP	wireless access point
WPA	Wired Protected Access

---