



## CLI GUIDE

# Cisco Small Business 300 1.1 Series Managed Switch Administration Guide

## Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>17</b>
<b>2</b>	<b>User Interface Commands .....</b>	<b>33</b>
	enable .....	33
	disable .....	34
	login .....	34
	configure .....	35
	exit (Configuration) .....	36
	exit (EXEC) .....	36
	end .....	37
	help .....	38
	history .....	39
	history size .....	40
	terminal history .....	41
	terminal history size .....	41
	terminal datadump .....	42
	show history .....	43
	show privilege .....	44
	do .....	45
	banner login .....	46
	login-banner .....	48
	show banner .....	49
<b>3</b>	<b>Macro Commands .....</b>	<b>50</b>
	macro name .....	50
	macro apply .....	53
	macro description .....	55
	macro global .....	57
	macro global description .....	58
	show parser macro .....	59
<b>4</b>	<b>RSA and Certificate Commands .....</b>	<b>62</b>
	crypto key generate dsa .....	62
	crypto key generate rsa .....	62
	show crypto key mypubkey .....	63
	crypto certificate generate .....	64
	crypto certificate request .....	66
	crypto certificate import .....	67
	show crypto certificate mycertificate .....	69
<b>5</b>	<b>System Management Commands .....</b>	<b>71</b>
	ping .....	71
	traceroute .....	74
	telnet .....	77
	resume .....	81
	hostname .....	82

	reload .....	82
	service cpu-utilization .....	83
	show cpu utilization .....	84
	show users .....	85
	show sessions .....	86
	show system .....	87
	show version .....	88
	show version md5 .....	89
	system resources routing .....	90
	show system resources .....	91
	set system mode .....	92
	show system mode .....	93
	show system languages .....	93
	show system tcam utilization .....	94
	show services tcp-udp .....	95
	show system id .....	96
	show cpu input rate .....	96
<b>6</b>	<b>Clock Commands .....</b>	<b>98</b>
	clock set .....	98
	clock source .....	98
	clock timezone .....	99
	clock summer-time .....	100
	clock dhcp timezone .....	102
	sntp authentication-key .....	103
	sntp authenticate .....	104
	sntp trusted-key .....	105
	sntp broadcast client enable .....	106
	sntp unicast client enable .....	107
	sntp server .....	107
	show clock .....	110
	show sntp configuration .....	111
	show sntp status .....	113
<b>7</b>	<b>Configuration and Image File Commands .....</b>	<b>115</b>
	copy .....	115
	write memory .....	120
	write .....	121
	delete .....	121
	dir .....	122
	more .....	124
	boot system .....	125
	show bootvar .....	126
	show running-config .....	126
	show startup-config .....	129
<b>8</b>	<b>Auto-Configuration .....</b>	<b>133</b>
	boot host auto-config .....	133

	show boot .....	133
	ip dhcp tftp-server ip address .....	134
	ip dhcp tftp-server file .....	135
	show ip dhcp tftp-server .....	135
	ip dhcp information option .....	136
	show ip dhcp information option .....	137
<b>9</b>	<b>Management ACL Commands.....</b>	<b>138</b>
	management access-list .....	138
	permit (Management) .....	139
	deny (Management) .....	140
	management access-class .....	142
	show management access-list .....	142
	show management access-class .....	143
<b>10</b>	<b>Network Management Protocol (SNMP) Commands.....</b>	<b>145</b>
	snmp-server .....	145
	snmp-server community .....	145
	snmp-server view .....	148
	snmp-server group .....	149
	snmp-server user .....	151
	snmp-server filter .....	153
	snmp-server host .....	154
	snmp-server engineID remote .....	156
	snmp-server enable traps .....	157
	snmp-server trap authentication .....	158
	snmp-server contact .....	159
	snmp-server location .....	159
	snmp-server set .....	160
	show snmp .....	161
	show snmp engineID .....	162
	show snmp views .....	163
	show snmp groups .....	164
	show snmp filters .....	165
	show snmp users .....	166
<b>11</b>	<b>Web Server Commands .....</b>	<b>167</b>
	ip http server .....	167
	ip http timeout-policy .....	167
	ip http secure-server .....	168
	ip https certificate .....	169
	show ip http .....	170
	show ip https .....	170
<b>12</b>	<b>Teletype Network (Telnet), Secure Shell (SSH) and Secure Login (Slogin) Commands....</b>	<b>172</b>
	ip telnet server .....	172
	ip ssh server .....	172
	user-key .....	173

	key-string .....	174
	crypto key pubkey-chain ssh .....	176
	show crypto key pubkey-chain ssh .....	177
	show ip ssh .....	178
<b>13</b>	<b>Line Commands.....</b>	<b>180</b>
	line .....	180
	speed .....	180
	autobaud .....	181
	exec-timeout .....	182
	show line .....	183
<b>14</b>	<b>Bonjour Commands.....</b>	<b>185</b>
	bonjour enable .....	185
	bonjour interface range .....	185
	show bonjour .....	186
<b>15</b>	<b>Authentication, Authorization and Accounting (AAA) Commands .....</b>	<b>188</b>
	aaa authentication login .....	188
	aaa authentication enable .....	189
	ip http authentication .....	191
	show authentication methods .....	192
	password .....	194
	enable password .....	195
	username .....	196
	show user accounts .....	197
	passwords complexity enable .....	198
	passwords complexity <attributes> .....	200
	passwords aging .....	202
	show passwords configuration .....	203
<b>16</b>	<b>Remote Authentication Dial-In User Service (RADIUS) Commands.....</b>	<b>205</b>
	radius-server host .....	205
	radius-server key .....	207
	radius-server retransmit .....	208
	radius-server source-ip .....	209
	radius-server source-ipv6 .....	210
	radius-server timeout .....	210
	radius-server deadtime .....	211
	show radius-servers .....	212
<b>17</b>	<b>Terminal Access Controller Access-Control System Plus (TACACS+) Commands ....</b>	<b>214</b>
	tacacs-server host .....	214
	tacacs-server key .....	215
	tacacs-server timeout .....	216
	tacacs-server source-ip .....	217
	show tacacs .....	217
<b>18</b>	<b>Syslog Commands .....</b>	<b>219</b>
	logging on .....	219

	logging host .....	220
	logging console .....	221
	logging buffered .....	222
	clear logging .....	223
	logging file .....	224
	clear logging file .....	225
	file-system logging .....	225
	logging aggregation on .....	226
	logging aggregation aging-time .....	227
	show logging .....	227
	show logging file .....	229
	show syslog-servers .....	230
<b>19</b>	<b>Remote Network Monitoring (RMON) Commands.....</b>	<b>232</b>
	show rmon statistics .....	232
	rmon collection stats .....	234
	show rmon collection stats .....	235
	show rmon history .....	236
	rmon alarm .....	239
	show rmon alarm-table .....	241
	show rmon alarm .....	242
	rmon event .....	244
	show rmon events .....	245
	show rmon log .....	246
	rmon table-size .....	247
<b>20</b>	<b>802.1x Commands.....</b>	<b>249</b>
	aaa authentication dot1x .....	249
	dot1x system-auth-control .....	250
	dot1x port-control .....	250
	dot1x reauthentication .....	252
	dot1x timeout reauth-period .....	252
	dot1x timeout quiet-period .....	253
	dot1x timeout tx-period .....	254
	dot1x max-req .....	255
	dot1x timeout supp-timeout .....	256
	dot1x timeout server-timeout .....	257
	show dot1x .....	258
	show dot1x users .....	262
	show dot1x statistics .....	263
	clear dot1x statistics .....	266
	dot1x host-mode .....	266
	dot1x violation-mode .....	267
	dot1x guest-vlan .....	269
	dot1x guest-vlan timeout .....	269
	dot1x guest-vlan enable .....	270
	dot1x mac-authentication .....	271
	show dot1x advanced .....	272

<b>21</b>	<b>Ethernet Configuration Commands</b>	<b>274</b>
	interface	274
	interface range	274
	shutdown	275
	description	276
	speed	276
	duplex	277
	negotiation	278
	flowcontrol	279
	mdix	280
	back-pressure	280
	port jumbo-frame	281
	clear counters	282
	set interface active	282
	show interfaces configuration	283
	show interfaces status	284
	show interfaces advertise	285
	show interfaces description	286
	show interfaces counters	287
	show port jumbo-frame	290
	storm-control broadcast level kbps	291
	storm-control broadcast level	291
	storm-control include-multicast	293
	show storm-control	293
<b>22</b>	<b>PHY Diagnostics Commands</b>	<b>295</b>
	show cable-diagnostics cable-length	295
	show fiber-ports optical-transceiver	295
<b>23</b>	<b>Power over Ethernet (PoE) Commands</b>	<b>298</b>
	power inline	298
	power inline priority	298
	power inline usage-threshold	299
	power inline traps enable	300
	power inline limit	300
	power inline limit-mode	301
	show power inline	302
	show power inline consumption	304
<b>24</b>	<b>EEE Commands</b>	<b>306</b>
	eee enable (global)	306
	eee enable (interface)	306
	eee lldp enable	307
	show eee	307
<b>25</b>	<b>Green Ethernet</b>	<b>313</b>
	green-ethernet energy-detect (global)	313
	green-ethernet energy-detect (interface)	313
	show green-ethernet	314

	green-ethernet short-reach (global) .....	316
	green-ethernet short-reach (interface) .....	316
	green-ethernet power-meter reset .....	317
<b>26</b>	<b>Port Channel Commands .....</b>	<b>319</b>
	port-channel load-balance .....	320
	show interfaces port-channel .....	320
<b>27</b>	<b>Address Table Commands .....</b>	<b>323</b>
	bridge multicast filtering .....	323
	bridge multicast mode .....	323
	bridge multicast address .....	325
	bridge multicast forbidden address .....	327
	bridge multicast ip-address .....	328
	bridge multicast forbidden ip-address .....	329
	bridge multicast source group .....	330
	bridge multicast forbidden source group .....	332
	bridge multicast ipv6 mode .....	333
	bridge multicast ipv6 ip-address .....	335
	bridge multicast ipv6 forbidden ip-address .....	336
	bridge multicast ipv6 source group .....	337
	bridge multicast ipv6 forbidden source group .....	338
	bridge multicast unregistered .....	340
	bridge multicast forward-all .....	341
	bridge multicast forbidden forward-all .....	342
	mac address-table static .....	343
	clear mac address-table .....	344
	mac address-table aging-time .....	344
	port security .....	345
	port security mode .....	346
	port security max .....	347
	show mac address-table .....	348
	show mac address-table count .....	349
	show bridge multicast mode .....	350
	show bridge multicast address-table .....	351
	show bridge multicast unregistered .....	354
	show ports security .....	355
	show ports security addresses .....	356
	bridge multicast reserved-address .....	357
	show bridge multicast reserved-addresses .....	358
<b>28</b>	<b>Port Monitor Commands .....</b>	<b>360</b>
	port monitor .....	360
	show ports monitor .....	362
<b>29</b>	<b>Spanning-Tree Commands .....</b>	<b>363</b>
	spanning-tree .....	363
	spanning-tree mode .....	363
	spanning-tree forward-time .....	364



spanning-tree hello-time .....	365
spanning-tree max-age .....	366
spanning-tree priority .....	367
spanning-tree disable .....	368
spanning-tree cost .....	368
spanning-tree port-priority .....	369
spanning-tree portfast .....	370
spanning-tree link-type .....	371
spanning-tree pathcost method .....	372
spanning-tree bpdu (Global) .....	373
spanning-tree bpdu (Interface) .....	374
spanning-tree bpduguard .....	375
clear spanning-tree detected-protocols .....	376
spanning-tree mst priority .....	376
spanning-tree mst max-hops .....	377
spanning-tree mst port-priority .....	378
spanning-tree mst cost .....	379
spanning-tree mst configuration .....	380
instance (MST) .....	381
name (MST) .....	382
revision (MST) .....	382
show (MST) .....	383
exit (MST) .....	384
abort (MST) .....	384
show spanning-tree .....	385
show spanning-tree bpdu .....	401
<b>30 Virtual Local Area Network (VLAN) Commands.....</b>	<b>403</b>
vlan database .....	403
vlan .....	403
show vlan .....	404
default-vlan vlan .....	406
show default-vlan-membership .....	407
interface vlan .....	408
interface range vlan .....	409
name .....	410
switchport protected-port .....	410
show interfaces protected-ports .....	411
switchport mode .....	412
switchport access vlan .....	413
switchport trunk allowed vlan .....	414
switchport trunk native vlan .....	415
switchport general allowed vlan .....	416
switchport general pvid .....	418
switchport general ingress-filtering disable .....	420
switchport general acceptable-frame-type .....	421
switchport customer vlan .....	422

	map mac macs-group .....	422
	switchport general map macs-group vlan .....	424
	show vlan macs-groups .....	425
	switchport forbidden default-vlan .....	426
	switchport forbidden vlan .....	426
	switchport default-vlan tagged .....	427
	show interfaces switchport .....	429
	ip internal-usage-vlan .....	432
	show vlan internal usage .....	433
<b>31</b>	<b>Internet Group Management Protocol (IGMP) Snooping Commands .....</b>	<b>435</b>
	ip igmp snooping (Global) .....	435
	ip igmp snooping vlan .....	435
	ip igmp snooping vlan mrouter .....	436
	ip igmp snooping vlan mrouter interface .....	437
	ip igmp snooping vlan forbidden mrouter interface .....	438
	ip igmp snooping vlan static .....	439
	ip igmp snooping vlan querier .....	440
	ip igmp snooping vlan querier address .....	441
	ip igmp snooping vlan querier version .....	442
	ip igmp robustness .....	442
	ip igmp query-interval .....	443
	ip igmp query-max-response-time .....	444
	ip igmp last-member-query-count .....	445
	ip igmp last-member-query-interval .....	446
	ip igmp snooping vlan immediate-leave .....	446
	show ip igmp snooping mrouter .....	447
	show ip igmp snooping interface .....	448
	show ip igmp snooping groups .....	449
<b>32</b>	<b>IPv6 MLD Snooping Commands.....</b>	<b>451</b>
	ipv6 mld snooping (Global) .....	451
	ipv6 mld snooping vlan .....	451
	ipv6 mld robustness .....	452
	ipv6 mld snooping mrouter .....	453
	ipv6 mld snooping mrouter interface .....	454
	ipv6 mld snooping forbidden mrouter interface .....	455
	ipv6 mld snooping static .....	455
	ipv6 mld query-interval .....	456
	ipv6 mld query-max-response-time .....	457
	ipv6 mld last-member-query-count .....	458
	ipv6 mld last-member-query-interval .....	459
	ipv6 mld snooping vlan immediate-leave .....	459
	show ipv6 mld snooping mrouter .....	460
	show ipv6 mld snooping interface .....	461
	show ipv6 mld snooping groups .....	462
<b>33</b>	<b>Link Aggregation Control Protocol (LACP) Commands .....</b>	<b>464</b>

	lacp system-priority .....	464
	lacp port-priority .....	464
	lacp timeout .....	465
	show lacp .....	466
	show lacp port-channel .....	469
<b>34</b>	<b>GARP VLAN Registration Protocol (GVRP) Commands .....</b>	<b>471</b>
	gvrp enable (Global) .....	471
	gvrp enable (Interface) .....	471
	gvrp vlan-creation-forbid .....	472
	gvrp registration-forbid .....	473
	clear gvrp statistics .....	473
	show gvrp configuration .....	474
	show gvrp statistics .....	475
	show gvrp error-statistics .....	476
<b>35</b>	<b>IP Addressing Commands.....</b>	<b>478</b>
	ip address .....	478
	ip address dhcp .....	479
	renew dhcp .....	481
	ip default-gateway .....	482
	show ip interface .....	482
	arp .....	483
	arp timeout (Global) .....	484
	ip arp proxy disable .....	485
	ip proxy-arp .....	486
	clear arp-cache .....	486
	show arp .....	487
	show arp configuration .....	488
	interface ip .....	489
	ip helper-address .....	490
	show ip helper-address .....	491
	ip domain name .....	492
	ip name-server .....	493
	ip host .....	494
	clear host .....	495
	clear host dhcp .....	496
	show hosts .....	497
<b>36</b>	<b>IPv6 Addressing Commands .....</b>	<b>499</b>
	ipv6 enable .....	499
	ipv6 address autoconfig .....	500
	ipv6 icmp error-interval .....	501
	show ipv6 icmp error-interval .....	502
	ipv6 address .....	502
	ipv6 address link-local .....	503
	ipv6 unreachable .....	504
	ipv6 default-gateway .....	505

	show ipv6 interface .....	506
	show IPv6 route .....	508
	ipv6 nd dad attempts .....	509
	ipv6 host .....	510
	ipv6 neighbor .....	511
	ipv6 set mtu .....	512
	show ipv6 neighbors .....	513
	clear ipv6 neighbors .....	515
<b>37</b>	<b>Tunnel Commands .....</b>	<b>516</b>
	interface tunnel .....	516
	tunnel mode ipv6ip .....	516
	tunnel isatap router .....	517
	tunnel source .....	518
	tunnel isatap query-interval .....	519
	tunnel isatap solicitation-interval .....	520
	tunnel isatap robustness .....	521
	show ipv6 tunnel .....	522
<b>38</b>	<b>DHCP Relay Commands.....</b>	<b>524</b>
	ip dhcp relay enable (Global) .....	524
	ip dhcp relay enable (Interface) .....	524
	ip dhcp relay address .....	525
	show ip dhcp relay .....	526
<b>39</b>	<b>IP Routing Protocol-Independent Commands.....</b>	<b>529</b>
	ip route .....	529
	show ip route .....	530
<b>40</b>	<b>ACL Commands.....</b>	<b>533</b>
	ip access-list .....	533
	permit ( IP ) .....	534
	deny ( IP ) .....	536
	ipv6 access-list .....	539
	permit ( IPv6 ) .....	540
	deny ( IPv6 ) .....	542
	mac access-list .....	544
	permit ( MAC ) .....	545
	deny ( MAC ) .....	546
	service-acl .....	548
	absolute .....	549
	periodic .....	550
	show time-range .....	551
	show access-lists .....	552
	show interfaces access-lists .....	554
	clear access-lists counters .....	554
	show interfaces access-lists counters .....	555
<b>41</b>	<b>Quality of Service (QoS) Commands .....</b>	<b>557</b>
	qos .....	557

qos advanced-mode trust .....	558
show qos .....	559
class-map .....	560
show class-map .....	561
match .....	562
policy-map .....	563
class .....	564
show policy-map .....	565
trust .....	566
set .....	568
police .....	569
service-policy .....	570
qos aggregate-policer .....	571
show qos aggregate-policer .....	572
police aggregate .....	573
wrr-queue cos-map .....	574
wrr-queue bandwidth .....	575
priority-queue out num-of-queues .....	577
traffic-shape .....	578
traffic-shape queue .....	578
rate-limit (Ethernet) .....	579
rate-limit (VLAN) .....	580
qos wrr-queue wrtd .....	581
show qos wrr-queue wrtd .....	582
show qos interface .....	583
wrr-queue .....	585
qos wrr-queue threshold .....	586
qos map policed-dscp .....	587
qos map dscp-queue .....	588
qos map dscp-dp .....	589
qos trust (Global) .....	590
qos trust (Interface) .....	591
qos cos .....	592
qos dscp-mutation .....	592
qos map dscp-mutation .....	593
show qos map .....	594
clear qos statistics .....	595
qos statistics policer .....	596
qos statistics aggregate-policer .....	596
qos statistics queues .....	597
show qos statistics .....	598
security-suite enable .....	600
security-suite dos protect .....	601
security-suite dos syn-attack .....	603
security-suite deny martian-addresses .....	604
security-suite deny syn .....	606
security-suite deny icmp .....	607

	security-suite deny fragmented .....	609
	show security-suite configuration .....	610
<b>42</b>	<b>Voice VLAN Commands.....</b>	<b>612</b>
	voice vlan state .....	612
	voice vlan refresh .....	614
	voice vlan id .....	615
	voice vlan vpt .....	616
	voice vlan dscp .....	617
	voice vlan oui-table .....	618
	voice vlan cos mode .....	620
	voice vlan cos .....	621
	voice vlan aging-timeout .....	621
	voice vlan enable .....	622
	show voice vlan .....	623
	show voice vlan local .....	626
<b>43</b>	<b>Smartport Commands .....</b>	<b>629</b>
	macro auto (Global) .....	629
	macro auto smartport (Interface) .....	630
	macro auto trunk refresh .....	631
	macro auto resume .....	632
	macro auto persistent .....	633
	macro auto smartport type .....	634
	macro auto processing cdp .....	636
	macro auto processing lldp .....	637
	macro auto processing type .....	638
	macro auto user smartport macro .....	639
	macro auto built-in parameters .....	640
	show macro auto processing .....	641
	show macro auto smart-macros .....	642
	show macro auto ports .....	643
	smartport switchport trunk allowed vlan .....	645
	smartport switchport trunk native vlan .....	646
<b>44</b>	<b>Link Layer Discovery Protocol (LLDP) Commands.....</b>	<b>648</b>
	lldp run .....	648
	lldp transmit .....	648
	lldp receive .....	649
	lldp timer .....	650
	lldp hold-multiplier .....	651
	lldp reinit .....	652
	lldp tx-delay .....	652
	lldp optional-tlv .....	653
	lldp management-address .....	654
	lldp notifications .....	655
	lldp notifications interval .....	656
	lldp lldpdu .....	657

lldp med .....	658
lldp med notifications topology-change .....	659
lldp med fast-start repeat-count .....	659
lldp med network-policy (global) .....	660
lldp med network-policy (interface) .....	662
lldp med network-policy voice auto .....	663
clear lldp table .....	664
lldp med location .....	665
show lldp configuration .....	666
show lldp med configuration .....	668
show lldp local tlvs-overloading .....	670
show lldp local .....	671
show lldp statistics .....	673
show lldp neighbors .....	674
<b>45 CDP Commands .....</b>	<b>681</b>
cdp run .....	681
cdp enable .....	682
cdp pdu .....	682
cdp advertise-v2 .....	683
cdp appliance-tlv enable .....	684
cdp mandatory-tlvs validation .....	685
cdp source-interface .....	686
cdp log mismatch duplex .....	686
cdp log mismatch voip .....	687
cdp log mismatch native .....	688
cdp device-id format .....	689
cdp timer .....	689
cdp holdtime .....	690
clear cdp counters .....	691
clear cdp table .....	691
show cdp .....	692
show cdp entry .....	693
show cdp interface .....	695
show cdp neighbors .....	696
show cdp tlv .....	700
show cdp traffic .....	704
<b>46 Revision History .....</b>	<b>706</b>

# 1 Introduction

## Overview

This chapter describes the CLI command modes, how to access the CLI, and the CLI command editing features.

## CLI Command Modes

The Command Line Interface (CLI) is divided into four command modes. The command modes are (in the order in which they are accessed):

- User EXEC mode
- Privileged EXEC mode
- Global Configuration mode
- Interface Configuration mode

Each command mode has its own unique console prompt and set of CLI commands. Entering a question mark at the console prompt displays a list of available commands for the current mode and for the level of the user. Specific commands are used to switch from one mode to another.

### User EXEC Mode

Users with level 1 initially log into User EXEC mode. User EXEC mode is used for tasks that do not change the configuration, such as performing basic tests and listing system information.

The user-level prompt consists of the Switch host name followed by the angle bracket >:

```
Console>
```

The default host name is **Console** unless it is changed via the **hostname** command in Global Configuration mode.



To access the next higher mode (Privileged EXEC mode), use the **enable** command and enter the password for the higher level when prompted.

## Privileged EXEC Mode

Users with levels 7 and 15 initially log into Privileged EXEC mode, which is password-protected to prevent unauthorized use, as many of its commands set operating system parameters. The password is not displayed on the screen and is case-sensitive.

Only users with command level of 7 or 15 can access this mode.

To enter this mode from User EXEC mode, follow these steps:

- 
- STEP 1** At the prompt enter the **enable** command and press **Enter**. A password prompt is displayed.
  - STEP 2** Enter the password to go the next level and press **Enter**. (For security purposes, each character in the password is replaced by \*). The Privileged EXEC mode prompt, consisting of the Switch host name followed by #, is displayed.

**Console#**

To return from the Privileged EXEC mode to the User EXEC mode, use the **disable** command.

The following example illustrates how to access Privileged EXEC mode and then return to User EXEC mode:

```
Console>  
Console> enable  
Enter Password: *****  
Console#  
Console# disable  
Console>
```

## Global Configuration Mode

The Global Configuration mode is used to configure features at the system level, and not at the interface level.

Only users with command level of 7 or 15 can access this mode.

To access Global Configuration mode from Privileged EXEC mode, enter the **configure** command at the Privileged EXEC mode prompt and press **Enter**. The Global Configuration mode prompt, consisting of the device host name followed by **(config)#**, is displayed:

```
Console(config)#
```

Use any of the following commands to return from Global Configuration mode to the Privileged EXEC mode:

```
exit
```

```
end
```

```
Ctrl+Z
```

The following example shows how to access Global Configuration mode and return to Privileged EXEC mode:

```
Console#  
Console# configure  
Console(config)# exit  
Console#
```

## Interface or Line Configuration Modes

Various submodes may be entered from Global Configuration mode. These submodes enable performing commands on a group of interfaces or lines.

For instance to perform several operations on a specific port or range of ports, you can enter the Interface Configuration mode for that interface.

The following example enters Interface Configuration mode for ports gi1-5 and then sets their speed:

```
Console#  
Console# configure  
Console(config)# interface range gil-5  
Console(config-if)#speed 10  
Console(config-if)#exit  
Console(config)#
```

The **exit** command returns to Global Configuration mode.

The following submodes are available:

- **Interface** — Contains commands that configure a specific interface (port, VLAN, port channel, or tunnel) or range of interfaces. The Global Configuration mode command `interface` is used to enter the Interface Configuration mode. The **interface** Global Configuration command is used to enter this mode.
- **Line Interface** — Contains commands used to configure the management connections for the console, Telnet and SSH. These include commands such as line timeout settings, etc. The **line** Global Configuration command is used to enter the Line Configuration command mode.
- **VLAN Database** — Contains commands used to configure a VLAN as a whole. The **vlan database** Global Configuration mode command is used to enter the VLAN Database Interface Configuration mode.
- **Management Access List** — Contains commands used to define management access-lists. The **management access-list** Global Configuration mode command is used to enter the Management Access List Configuration mode.
- **Port Channel** — Contains commands used to configure port-channels; for example, assigning ports to a port-channel. Most of these commands are the same as the commands in the Ethernet interface mode, and are used to manage the member ports as a single entity. The **interface port-channel** Global Configuration mode command is used to enter the Port Channel Interface Configuration mode.
- **QoS** — Contains commands related to service definitions. The **qos** Global Configuration mode command is used to enter the QoS services configuration mode.

- **MAC Access-List** — Configures conditions required to allow traffic based on MAC addresses. The **mac access-list** Global Configuration mode command is used to enter the MAC access-list configuration mode.

To return from any Interface Configuration mode to the Global Configuration mode, use the **exit** command.

## User (Privilege) Levels

Users may be created with one of the following user levels:

- **Level 1** — Users with this level can only run User EXEC mode commands. Users at this level cannot access the web GUI.
- **Level 7** — Users with this level can run commands in the User EXEC mode and a subset of commands in the Privileged EXEC mode. Users at this level cannot access the web GUI.
- **Level 15** — Users with this level can run all commands. Only users at this level can access the web GUI.

A system administrator (user with level 15) can create passwords that allow a lower level user to temporarily become a higher level user. For example, the user may go from level 1 to level 7, level 1 to 15, or level 7 to level 15.

The passwords for each level are set (by an administrator) using the following command:

```
enable password [level privilege-level] {password | encrypted  
encrypted-password}
```

Users with a lower level can raise their level by entering the command: **enable** and the password for level 7 or 15. A user can go from level 1 to level 7 or directly to level 15. The higher level holds only for the current session.

The **disable** command returns the user to a lower level.

To create a user and assign it a user level, use the **username** command. Only users with command level 15, can create users at this level.

Example — Create passwords for level 7 and 15 (by the administrator)

```
Console>configure
Console<conf># enable password level 7 level7@abc
Console<conf># enable password level 15 level15@abc
Console<conf>#
```

Create a user with user level 1:

```
Console>
Console> username john password john1234 privilege 1
Console>
```

Example 2— Switch between Level 1 to Level 15. The user must know the password.

```
Console>
Console> enable
Enter Password: ***** (this is the password for level 15
- level15@abc)
Console>
```



**NOTE** If authentication of passwords is performed on RADIUS or TACACS+ servers, the passwords assigned to user level 7 and user level 15 must be configured on the external server and associated with the \$enable7\$ and \$enable15\$ user names, respectively. See the [Authentication, Authorization and Accounting \(AAA\) Commands](#) chapter for details.

## Accessing the Command Line Interface

The Command Line Interface (CLI) can be accessed from a terminal or computer by performing one of the following tasks:

- Running a terminal application, such as HyperTerminal, on a computer that is directly connected to the Switch's console port,
- or—
- Running a Telnet session from a command prompt on a computer with a network connection to the Switch.
- Using SSH.



---

**NOTE** Telnet and SSH are disabled by default on the switch.

---

If access is via a Telnet connection, ensure that the following conditions are met before using CLI commands:

- The switch has a defined IP address.
- Corresponding management access is granted.
- There is an IP path such that the computer and the switch can reach each other.

### Using HyperTerminal over the Console Interface



---

**NOTE** When using HyperTerminal with Microsoft® Windows® 2000, ensure that Windows® 2000 Service Pack 2 or later is installed on your computer. The arrow keys will not function properly using HyperTerminal's VT100 emulation in Windows® 2000 prior to Service Pack 2. For information on Windows® 2000 service packs, go to [www.microsoft.com](http://www.microsoft.com).

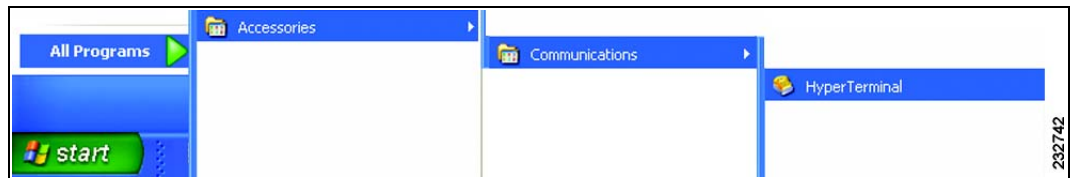
---

The Switch's RS-232 serial console port provides a direct connection to a computer's serial port using a standard DB-9 null-modem or crossover cable. Once the computer and Switch are connected, run a terminal application to access the Command Line Interface.

To access the Command Line Interface using the HyperTerminal application, perform the following steps:

- STEP 1** Click the **Start** button.
- STEP 2** Select **All Programs > Accessories > Communications > HyperTerminal**.

**Figure 1 Start > All Programs > Accessories > Communications > HyperTerminal**



- STEP 3** Enter a name for this connection. Select an icon for the application, then click **OK**.
- STEP 4** Select a port to communicate with the switch. Select **COM1** or **COM2**.
- STEP 5** Set the serial port settings, then click **OK**.
- STEP 6** When the *Command Line Interface* appears, enter **admin** at the *User Name* prompt and press **Enter**.

**Figure 2 Command Line User Name Prompt**



The **console#** prompt is displayed. This prompt is where you enter CLI commands.

**Figure 3 Command Line**



You can now enter CLI commands to manage the switch. For detailed information on CLI commands, refer to the appropriate chapter(s) of this Reference Guide.

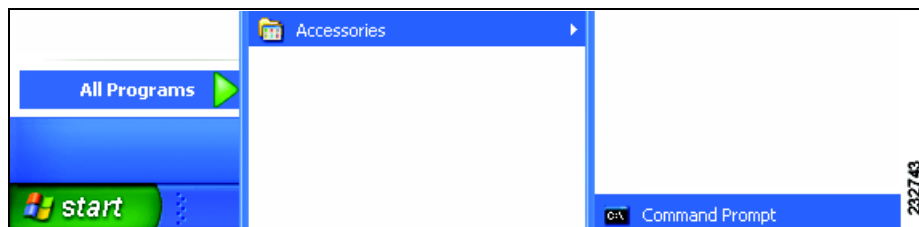
## Using Telnet over an Ethernet Interface

Telnet provides a method of connecting to the Command Line Interface over an IP network.

To establish a telnet session from the command prompt, perform the following steps:

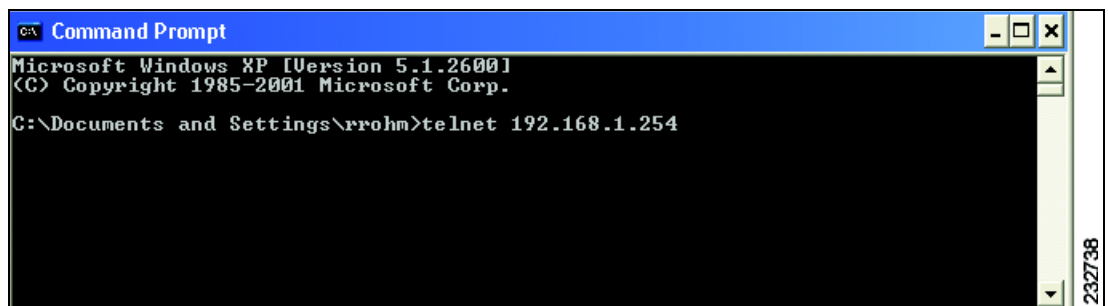
- STEP 1** Click **Start**, then select **All Programs > Accessories > Command Prompt** to open a command prompt.

**Figure 4 Start > All Programs > Accessories > Command Prompt**



- STEP 2** At the prompt, enter **telnet 1<IP address of switch>**, then press **Enter**.

**Figure 5 Command Prompt**



- STEP 3** The *Command Line Interface* will be displayed.



## Editing Features

### Entering Commands

A CLI command is a series of keywords and arguments. Keywords identify a command, and arguments specify configuration parameters. For example, in the command **show interfaces status GigabitEthernet 1**, *show*, *interfaces* and *status* are keywords, *GigabitEthernet* is an argument that specifies the interface type, and *1* specifies the port.

To enter commands that require parameters, enter the required parameters after the command keyword. For example, to set a password for the administrator, enter:

```
Console(config)# username admin password alansmith
```

When working with the CLI, the command options are not displayed. The standard command to request help is `?`.

There are two instances where help information can be displayed:

- **Keyword lookup** — The character `?` is entered in place of a command. A list of all valid commands and corresponding help messages are displayed.
- **Partial keyword lookup** — If a command is incomplete and or the character `?` is entered in place of a parameter, the matched keyword or parameters for this command are displayed.

To assist in using the CLI, there is an assortment of editing features. The following features are described:

- **Terminal Command Buffer**
- **Command Completion**
- **Interface Naming Conventions**
- **Keyboard Shortcuts**

### Terminal Command Buffer

Every time a command is entered in the CLI, it is recorded on an internally managed Command History buffer. Commands stored in the buffer are maintained on a First In First Out (FIFO) basis. These commands can be recalled, reviewed, modified, and reissued. This buffer is not preserved across device resets.

Keyword	Description
Up-Arrow key Ctrl+P	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Down-Arrow key	Returns to more recent commands in the history buffer after recalling commands with the up-arrow key. Repeating the key sequence will recall successively more recent commands.

By default, the history buffer system is enabled, but it can be disabled at any time. For more information on enabling or disabling the history buffer, refer to the **history** command.

There is a standard default number of commands that are stored in the buffer. The standard number of 10 commands can be increased to 216. By configuring 0, the effect is the same as disabling the history buffer system. For more information on configuring the command history buffer, refer to the **history size** command.

To display the history buffer, refer to the **show history** command.

## Negating the Effect of Commands

For many configuration commands, the prefix keyword **no** can be entered to cancel the effect of a command or reset the configuration to the default value. This Reference Guide provides a description of the negation effect for each CLI command.

## Command Completion

If the command entered is incomplete, invalid or has missing or invalid parameters, then the appropriate error message is displayed. This assists in entering the correct command. By pressing **Tab** after an incomplete command is entered, the system will attempt to identify and complete the command. If the characters already entered are not enough for the system to identify a single matching command, press **?** to display the available commands matching the characters already entered.

## Interface Naming Conventions

### Interface ID

Within the CLI, interfaces are denoted by concatenating the following elements:

- **Type of interface:** The following types of interfaces are found on the various types of devices:
  - **Fast Ethernet (10/100 bits)** - This can be written as **FastEthernet** or **fa**.
  - **Gigabit Ethernet ports (10/100/1000 bits)** - This can be written either **Gigabit Ethernet** or **gi** or **GE**.
  - **LAG (Port Channel)** - This can be written as either **Port-Channel** or **po**.
  - **VLAN** - This is written as **VLAN**
  - **Tunnel** - This is written as **tunnel** or **tu**
- **Number of interface: Number of port, LAG, tunnel or VLAN**

The syntax for this is:

```
{<port-type>[ ]<port-number>}|{port-channel | po }[ ]<port-channel-number> | {tunnel | tu}[ ]<tunnel-number> |
vlan[ ]<vlan-id>
```

Sample of these various options are shown in the example below:

```
console#configure
console(config)#interface GigabitEthernet 1
console(config)#interface GE 1
console(config-if)#interface gil
console(config)#interface FastEthernet 1
console(config)#interface fa1
console(config-if)#interface po1
console(config-if)# interface vlan 1
```

## Interface Range

Interfaces may be described on an individual basis or within a range. The interface range command has the following syntax:

```
<interface-range> ::=
{<port-type>[ ] [<first-port-number>[ - <last-port-number>]} |
port-channel[ ] <first-port-channel-number>[ -
<last-port-channel-number>] |
tunnel[ ] <first-tunnel-number>[ - <last-tunnel-number>] |
vlan[ ] <first-vlan-id>[ - <last-vlan-id>]
```

A sample of this command is shown in the example below:

```
console#configure
console(config-if)#interface range gil-5
```

## Interface List

A combination of interface types can be specified in the **interface range** command in the following format:

```
<range-list> ::= <interface-range> | <range-list>, <
interface-range>
```

Up to five ranges can be included.



**NOTE** Range lists can contain either ports and port-channels or VLANs. Combinations of port/port-channels and VLANs are not allowed

The space after the comma is optional.

When a range list is defined, a space after the first entry and before the comma (,) must be entered.

A sample of this command is shown in the example below:

```
console#configure
cconsole(config-if)#interface range gi1-5, vlan 1-2
```

## Keyboard Shortcuts

The CLI has a range of keyboard shortcuts to assist in editing the CLI commands. The following table describes the CLI shortcuts.

Keyboard Key	Description
Up-arrow	Recalls commands from the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Down-arrow	Returns the most recent commands from the history buffer after recalling commands with the up arrow key. Repeating the key sequence will recall successively more recent commands.
Ctrl+A	Moves the cursor to the beginning of the command line.
Ctrl+E	Moves the cursor to the end of the command line.
Ctrl+Z / End	Returns back to the Privileged EXEC mode from any configuration mode.
Backspace	Deletes one character left to the cursor position.

## CLI Command Conventions

When entering commands there are certain command entry standards that apply to all commands. The following table describes the command conventions.

Convention	Description
[ ]	In a command line, square brackets indicate an optional entry.
{ }	In a command line, curly brackets indicate a selection of compulsory parameters separated the   character. One option must be selected. For example, <b>flowcontrol {autol on off}</b> means that for the flowcontrol command, either auto, on, or off must be selected.
<i>parameter</i>	Italic text indicates a parameter.
press <b>key</b>	Names of keys to be pressed are shown in <b>bold</b> .
Ctrl+F4	Keys separated by the + character are to be pressed simultaneously on the keyboard
Screen Display	Fixed-width font indicates CLI prompts, CLI commands entered by the user, and system messages displayed on the console.
all	When a parameter is required to define a range of ports or parameters and all is an option, the default for the command is <b>all</b> when no parameters are defined. For example, the command <b>interface range port-channel</b> has the option of either entering a range of channels, or selecting all. When the command is entered without a parameter, it automatically defaults to all.

## Copying and Pasting Text

Up to 1000 lines of text (or commands) can be copied and pasted into the device.



**NOTE** It is the user's responsibility to ensure that the text copied into the device consists of legal commands only.

When copying and pasting commands from a configuration file, make sure that the following conditions exist:

- A device Configuration mode has been accessed.

- The commands contain no encrypted data, like encrypted passwords or keys. Encrypted data cannot be copied and pasted into the device except for encrypted passwords where the keyword `encrypted` is used before the encrypted data (for instance in the `enable password` command).

## Layer 2 and Layer 3

The switch can operate in Switch mode (Layer 2) or Router mode (Layer 3).

The default mode is Switch mode (Layer 2 Mode). To change the mode of the switch, use the following CLI command:

```
set system mode {router | switch}
```

This command performs a system reboot.

In Layer 2 mode, the switch forwards packets as a VLAN-aware bridge. In Layer 3 mode, the switch performs both IPv4 routing and VLAN-aware bridging.

If Layer 2 mode is selected, a single IP address is supported on the default VLAN. The user also must configure a default gateway.

If Layer 3 mode is selected, the user can manage the device on any IP interface configured on the device, as long as a default route is configured. In Layer 3 mode, the switch routes traffic between IP VLANs, and bridges traffic with VLANs.

When the switch operates in Layer 3 mode, the following features are not supported:

- Protocol-based VLANs
- MAC-based VLANs
- VLAN Rate Limit
- DVA, Multicast TV VLAN
- Per flow policing

---

## 2 User Interface Commands

---

### 2.1 enable

The **enable** EXEC mode command enters the Privileged EXEC mode.

#### Syntax

**enable** [*privilege-level*]

#### Parameters

**privilege-level**—Specifies the privilege level at which to enter the system. (Range: 1, 7, 15)

#### Default Configuration

The default privilege level is 15.

#### Command Mode

EXEC mode

#### Example

The following example enters privilege level 7.

---

```
Console> enable 7
enter password:*****
Console#Accepted
```

---

The following example enters privilege level 15.

---

```
Console> enable
enter password:*****
Console#Accepted
```

---



---

## 2.2 disable

The **disable** Privileged EXEC mode command leaves the Privileged EXEC mode and returns to the User EXEC mode.

### Syntax

**disable** [*privilege-level*]

### Parameters

**privilege-level**—Reduces the privilege level to the specified privileged level. If privilege level is left blank, the level is reduce to 1.

### Default Configuration

The default privilege level is 1.

### Command Mode

Privileged EXEC mode

### Example

The following example returns the user to user level 7.

---

```
Console# disable 7
Console>
```

---

## 2.3 login

The **login** EXEC mode command enables changing the user that is logged in. When this command is logged in, the user is prompted for a username/password.

### Syntax

**login**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

EXEC mode

### Example

The following example enters Privileged EXEC mode and logs in with username 'admin'.

---

```
Console> login
User Name:admin
Password:*****
Console#
```

---

## 2.4 configure

The **configure** Privileged EXEC mode command enters the Global Configuration mode.

### Syntax

**configure** [*terminal*]

### Parameters

**terminal**—Enter the Global Configuration mode with or without the keyword terminal.

### Command Mode

Privileged EXEC mode

### Example

The following example enters Global Configuration mode.

---

```
Console# configure
Console(config)#
```

---

## 2.5 exit (Configuration)

The **exit** command exits any configuration mode to the next highest mode in the CLI mode hierarchy.

### Syntax

**exit**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

All commands in configuration modes.

### Examples

The following examples change the configuration mode from Interface Configuration mode to Privileged EXEC mode.

---

```
Console(config-if)# exit
```

```
Console(config)# exit
```

---

## 2.6 exit (EXEC)

The **exit** EXEC mode command closes an active terminal session by logging off the device.

### Syntax

**exit**

### Parameters

N/A

**Default Configuration**

N/A

**Command Mode**

EXEC mode

**Example**

The following example closes an active terminal session.

---

```
Console> exit
```

---

**2.7 end**

The **end** command ends the current configuration session and returns to the Privileged EXEC mode.

**Syntax****end****Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

All configuration modes

**Example**

The following example ends the Global Configuration mode session and returns to the Privileged EXEC mode.

---

```
Console(config)# end
```

```
Console#
```

---

## 2.8 help

The **help** command displays a brief description of the Help system.

### Syntax

**help**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

All command modes

### Example

The following example describes the Help system.

---

```
Console# help
```

```
Help may be requested at any point in a command by entering a question mark '?' . If nothing matches the currently entered incomplete command, the help list is empty. This indicates that there is no command matching the input as it currently appears. If the request is within a command, press the Backspace key and erase the entered characters to a point where the request results in a match.
```

```
Help is provided when:
```

1. There is a valid command and a help request is made for entering a parameter or argument (e.g. 'show ?'). All possible parameters or arguments for the entered command are then displayed.
2. An abbreviated argument is entered and a help request is made for arguments matching the input (e.g. 'show pr?').

---

## 2.9 history

The **history** Line Configuration mode command enables saving commands that have been entered. Use the **no** form of this command to disable the command.

### Syntax

**history**

**no history**

### Parameters

N/A

### Default Configuration

Enabled.

### Command Mode

Line Configuration mode

### User Guidelines

This command enables saving user-entered commands for a specified line. You can return to previous lines by using the up or down arrows.

The following are related commands:

- Use the [terminal history size](#) EXEC mode command to enable or disable this command for the current terminal session.
- Use the [history size](#) Line Configuration mode command to set the number of commands that are saved in the history.

### Example

The following example enables the command for Telnet.

---

```
Console(config)# line telnet
```

```
Console(config-line)# history
```

---

## 2.10 history size

The **history size** Line Configuration mode command changes the maximum number of user commands that are saved in the history buffer for a particular line. Use the **no** form of this command to reset the command history buffer size to the default value.

### Syntax

**history size** *number-of-commands*

**no history size**

### Parameters

**number-of-commands**—Specifies the number of commands the system records in its history buffer. (Range: 10–207)

### Default Configuration

The default command history buffer size is 10 commands.

### Command Mode

Line Configuration mode

### User Guidelines

This command configures the command history buffer size for a particular line. Use the **terminal history size** EXEC mode command to configure the command history buffer size for the current terminal session.

The allocated command history buffer is per terminal user, and is taken from a shared buffer. If there is not enough space available in the shared buffer, the command history buffer size cannot be increased above the default size.

### Example

The following example changes the command history buffer size to 100 entries for Telnet.

---

```
Console(config)# line telnet
```

```
Console(config-line)# history size 100
```

---

## 2.11 terminal history

The **terminal history** EXEC mode command enables the command history function for the current terminal session, meaning it is not stored in the Running Configuration file. Use the **no** form of this command to disable the command.

### Syntax

**terminal history**

**terminal no history**

### Default Configuration

The default configuration for all terminal sessions is defined by the [history](#) Line Configuration mode command.

### Command Mode

EXEC mode

### User Guidelines

The command enables the command history for the current session. The default is determined by the [history](#) Line Configuration mode command.

### Example

The following example disables the command history function for the current terminal session.

---

```
Console> terminal no history
```

---

## 2.12 terminal history size

The **terminal history size** EXEC mode command changes the command history buffer size for the current terminal session, meaning it is not stored in the Running Configuration file. Use the **no** form of this command to reset the command history buffer size to the default value.

### Syntax

**terminal history size** *number-of-commands*

**terminal no history size**



### Parameters

**number-of-commands**—Specifies the number of commands the system maintains in its history buffer. (Range: 10–207)

### Default Configuration

The default configuration for all terminal sessions is defined by the [history size](#) Line Configuration mode command.

### Command Mode

EXEC mode

### User Guidelines

The **terminal history size** EXEC command changes the command history buffer size for the current terminal session. Use the [history](#) Line Configuration mode command to change the default history buffer size.

The maximum number of commands in all buffers is 207.

### Example

The following example sets the command history buffer size to 20 commands for the current terminal session.

---

```
Console> terminal history size 20
```

---

## 2.13 terminal datadump

The **terminal datadump** EXEC mode command enables dumping all the output of a show command without prompting. Use the **no** form of this command to disable dumping.

### Syntax

**terminal datadump**

**no terminal datadump**

### Parameters

N/A

### Default Configuration

When printing, dumping is disabled and printing is paused every 24 lines.

### Command Mode

EXEC mode

### User Guidelines

By default, a **More** prompt is displayed when the output contains more than 24 lines. Pressing the **Enter** key displays the next line; pressing the **Spacebar** displays the next screen of output.

The **terminal datadump** command enables dumping all output immediately after entering the show command by removing the pause.

The width is currently not limited (previously the limit was 77 chars), and the width of the line being printed on the terminal is based on the terminal itself.

This command is relevant only for the current session.

### Example

The following example dumps all output immediately after entering a show command.

---

```
Console> terminal datadump
```

---

## 2.14 show history

The **show history** EXEC mode command lists commands entered in the current session.

### Syntax

**show history**

### Parameters

N/A

### Default Configuration

N/A

## Command Mode

EXEC mode

## User Guidelines

The buffer includes executed and unexecuted commands.

Commands are listed from the first to the most recent command.

The buffer remains unchanged when entering into and returning from configuration modes.

## Example

The following example displays all the commands entered while in the current Privileged EXEC mode.

---

```
Console# show version
SW version 3.131 (date 23-Jul-2005 time 17:34:19)
HW version 1.0.0
Console# show clock
15:29:03 Jun 17 2005
Console# show history
show version
show clock
show history
3 commands were logged (buffer size is 10)
```

---

## 2.15 show privilege

The **show privilege** EXEC mode command displays the current privilege level.

### Syntax

**show privilege**

### Parameters

N/A

**Default Configuration**

N/A

**Command Mode**

EXEC mode

**Example**

The following example displays the privilege level for the user logged on.

---

```
Console# show privilege
Current privilege level is 15
```

**2.16 do**

The **do** command executes an EXEC-level command from Global Configuration mode or any configuration submode.

**Syntax****do** *command***Parameters**

**command**—Specifies the EXEC-level command to execute.

**Command Mode**

All configuration modes

**Example**

The following example executes the **show vlan** Privileged EXEC mode command from Global Configuration mode.

**Example**


---

```
Console (Config)# do show vlan
```

Vlan	Name	Ports	Type	Authorization
----	----	-----	-----	-----

1	1	gi1-39,Po1,Po2,	other	Required
2	2	gi1	dynamicGvrp	Required
10	v0010	gi1	permanent	Not Required
11	V0011	gi1,gi3	permanent	Required
20	20	gi1	permanent	Required
30	30	gi1,gi3	permanent	Required
31	31	gi1	permanent	Required
91	91	gi1,gi4	permanent	Required
4093	guest-vlan	gi1,gi3	permanent	Guest

console(config)#

## 2.17 banner login

Use the **banner login** command in Global Configuration mode to specify a message to be displayed before the username and password login prompts. This banner is applied automatically on all the CLI interfaces: Console, Telnet and SSH and also on the WEB GUI. Use the **no** form of this command to delete the existing login banner.

### Syntax

**banner login** *d message-text d*

**no banner login**

### Parameters

- **d**—Delimiting character of user's choice—a pound sign (**#**), for example. You cannot use the delimiting character in the banner message.
- **message-text**—Message text. The message must start on a new line. You can enter multi-line messages. You can include tokens in the form of **\$(token)** in the message text. Tokens are replaced with the corresponding configuration variable (see User Guidelines). The message can contain up to 1000 characters (after every 510 characters, you must press <Enter> to continue).

### Default Configuration

Disabled (no Login banner is displayed).

## Command Mode

Global Configuration mode

### User Guidelines

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

Use tokens in the form of **\$(token)** in the message text to customize the banner. The tokens are described in the table below:

Token	Information displayed in the banner
\$(hostname)	Displays the host name for the device.
\$(domain)	Displays the domain name for the device.
\$(bold)	Indicates that the next text is a bold text. Using this token again indicates the end of the bold text.
\$(inverse)	Indicates that the next text is an inverse text. Using this token again indicates the end of the inverse text.
\$(contact)	Displays the system contact string.
\$(location)	Displays the system location string.
\$(mac-address)	Displays the base MAC address of the device.

Use the **no banner login** Line Configuration command to disable the Login banner on a particular line or lines.

### Example

The following example sets a Login banner that uses tokens. The percent sign (%) is used as a delimiting character. Note that the **\$(token)** syntax is replaced by the corresponding configuration variable.

```
Device(config)# banner login %
Enter TEXT message. End with the character '%'.
You have entered $(hostname).$(domain)
%
```

When the login banner is executed, the user will see the following banner:

```
You have entered host123.ourdomain.com
```

---

## 2.18 login-banner

Use the **login-banner** command in Line Configuration mode to enable the display of login banners. Use the **no** form of this command to disable the display of login banners.

### Syntax

**login-banner**

**no login-banner**

### Parameters

N/A

### Default Configuration

Enabled

### Command Mode

Line Configuration mode

### Example

---

```
console# configure
console(config)# line console
console(config-line)# login-banner
console(config-line)# exit
console(config)# line telnet
console(config-line)# login-banner
console(config-line)# exit
console(config)# line ssh
console(config-line)# login-banner
```

---

## 2.19 show banner

Use the **show banner** commands in EXEC mode to display the banners that have been defined.

### Syntax

**show banner login**

### Parameters

N/A

### Command Mode

EXEC mode

### Examples

---

```
console# show banner login
```

```
-----  
Banner: Login
```

```
Line SSH: Enabled
```

```
Line Telnet: Enabled
```

```
Line Console: Enabled
```



## 3 Macro Commands

### 3.1 macro name

There are two types of macros that can be created with the **macro name** Global Configuration mode command:

- Global macros are macros that are simple a group of CLI commands that can be used at any time.
- Smartport macros are associated with Smartport types (see [Section 43 "Smartport Commands"](#)). For each Smartport macro there must be an anti macro that concatenates **no\_** with the name of the Smartport macro.

If a macro by the name you assign, already exists, this macro overrides the previously-defined one.

Use the **no** form of this command to delete the macro definition.

#### Syntax

**macro name** *[macro-name]*

**no macro name** *[macro-name]*

#### Parameters

**macro-name**—Name of the macro. Macro names are case sensitive.

#### Default Configuration

The command has no default setting.

#### Command Mode

Global Configuration mode

#### User Guidelines

A macro is a script that contains CLI commands and is assigned a name by the user. It can contain up to 3000 characters and 200 lines.

A macro can contain up to three keywords/parameters. A keyword is a name prefixed with a '\$' character, for example, \$a\_keyword. The keywords in the macro's CLI commands are substituted with the values specified when you invoke the macro by using the **macro apply/trace** and **macro global apply/trace** CLI commands.

Use the following guidelines to create a macro:

- Use the **macro name** command to assign a name to the macro.
- Enter one macro command per line.
- Use the @ character to end the macro.
- Use the # character at the beginning of a line to enter comment text within the macro. In addition, # is used to identify certain preprocessor commands that can only be used within a macro. There are two possible preprocessor commands:
  - **#macro key description** - Each macro can be configured with up to 3 keyword and description pairs. When a macro is displayed from the GUI, its keywords and descriptions are displayed (if they exist).

The syntax for this preprocessor command is as follows:

- **#macro key description:** *\$keyword1. description\_string*
- **#\$keyword2:** *description\_string*
- **#\$keyword3:** *description\_string*

Parameters:

- **keyword** - A keyword must be prefixed with '\$'
- **description-string** - description of the keyword
- **#macro keywords** - This preprocessor command accepts up to 3 keywords. The command creates a CLI help string with the keywords for the macro. The help string will be displayed if help on the macro is requested from the **macro apply/trace** and **macro global apply trace** commands. The GUI also uses the keywords specified in the command as the parameter names for the macro. See Example 2 and 3 below for a description of how this command is used in the CLI.

The syntax for this preprocessor command is as follows:

- **#macro keywords:** *\$keyword-name1 \$keyword-name2*

Parameter:

- **keyword-name** - User-defined name of the keyword (parameter) prefixed with '\$'

It is important to consider the scope of any user-defined macro. Because of the potential hazards of applying unintended configurations, do not change

configuration modes within the macro by using commands such as **exit**, **end**, or **interface** *interface-id*. With few exceptions, there are other ways of executing macros in the various configuration modes.

You can modify a macro by creating a new macro with the same name as the existing macro. The newer macro overwrites the existing macro. The exceptions to this are the macros and corresponding anti-macros for Smartport. For example, you cannot edit the **host** and the **no\_host** macros.

### Examples

**Example 1** - The following example shows how to create a macro that when applied configures the duplex mode and speed of a port.

---

```
Switch(config)# macro name dup
Enter macro commands one per line. End with the character '@'.
macro description dup
no negotiation
duplex full
negotiation
@
```

---

**Example 2** - The following example shows how to create the same macro as in Example 1, but in this example, the macro has the parameters: DUPLEX and SPEED. When the macro is run, the values of DUPLEX and SPEED must be provided by the user. The **macro keywords** command enables the user to receive help for the macro as shown in Example 3.

---

```
Switch(config) # macro name duplex
Enter macro commands one per line. End with the character '@'.
duplex $DUPLEX
speed $SPEED
#macro keywords $DUPLEX $SPEED
@
```

---

**Example 3** - The following example shows how to display the keywords using the help character ? (as defined by the **macro keywords** command above):

---

```
switch<config-if># macro apply duplex ?
```

---

```

WORD <1-32> Keyword to replace with value e.g. $DUPLEX, $SPEED

<cr>

Switch<config-if> # macro apply duplex $DUPLEX ?

WORD<1-32> First parameter value

<cr>

Switch<config-if> # macro apply duplex $DUPLEX full $SPEED ?

WORD<1-32> Second parameter value

```

**Example 4** - The following example shows how to set the duplex mode of port gi1 to full and to set its speed to 100 Mb/s.

```

switch<config-if>#interface gi1

switch<config-if># macro apply duplex $DUPLEX full $SPEED 100

```

## 3.2 macro apply

Use the **macro apply/trace** Interface Configuration command to either:

- Apply a macro to an interface without displaying the actions being performed
- Apply a macro to the interface while displaying the actions being performed

### Syntax

```
macro {apply | trace} macro-name [parameter-name1 {value}] [parameter-name2 {value}] [parameter-name3 {value}]
```

### Parameters

- **apply**—Apply a macro to the specific interface.
- **trace**—Apply and trace a macro to the specific interface.
- **macro-name**—Name of the macro.
- ***parameter-name value***—(Optional) For each parameter defined in the macro, specify its name and value. You can enter up to three parameter-value pairs. Parameter keyword matching is case sensitive. All matching occurrences of the parameter name in the macro are replaced with the corresponding value.

### Default Configuration

The command has no default setting.

### Command Mode

Interface Configuration mode

### User Guidelines

The **macro apply** Interface Configuration mode command hides the commands of the macro from the user while it is being run. For debugging purposes, the **macro trace** command displays the commands along with any errors which are generated by them as they are executed." In this way, you can debug the macro and find syntax or configuration errors.

When you run a macro, if commands in it fail because of a syntax or configuration error, the macro continues to apply the remaining commands to the interface.

If you apply a macro that contains parameters in its commands, the command fails if you do not provide the values for the parameters. You can use the **macro apply** *macro-name* with a '?' to display the help string for the macro keywords (if you have defined these with the **#macro keywords** preprocessor command).

Parameter (keyword) matching is case sensitive. All matching occurrences of the parameter are replaced with the provided value. Any full match of a keyword, even if it is part of a large string, is considered a match and replaced by the corresponding value.

When you apply a macro to an interface, the switch automatically generates a macro description command with the macro name. As a result, the macro name is appended to the macro history of the interface. You can display the macro history of an interface using the **show parser macro description** command.

A macro applied to an interface range behaves the same way as a macro applied to a single interface. When a macro is applied to an interface range, it is applied sequentially to each interface within the range. If a macro command fails on one interface, it is nonetheless applied to the remaining interfaces.

### Examples.

---

**Example 1** - The following is an example of a macro being applied to an interface with the trace option.

```
Switch(config) # interface gi2

Switch<config-if> # macro trace dup

Applying command.. 'duplex full'
```

```
Applying command... `speed 100`  
Switch<config-if> #
```

---

**Example 2** - The following is an example of a macro being applied without the trace option.

```
Switch(config) # interface gi2  
Switch<config-if> # macro apply duplex $DUPLEX full $SPEED 100  
Switch<config-if> #
```

---

**Example 3** - The following is an example of an incorrect macro being applied.

```
switch(config-if)#macro trace dup  
Applying command...'duplex full'  
Applying command...'speed auto'  
% bad parameter value
```

---

### 3.3 macro description

Use the **macro description** Interface Configuration mode command to append a description, for example, a macro name, to the macro history of an interface. Use the **no** form of this command to clear the macro history of an interface. When the macro is applied to an interface, the switch automatically generates a macro description command with the macro name. As a result, the name of the macro is appended to the macro history of the interface.

#### Syntax

**macro description** *text*

**no macro description**

#### Parameters

**text**—Description text. The text can contain up to 160 characters. The text must be double quoted if it contains multiple words.

#### Default Configuration

The command has no default setting.

## Command Mode

Interface Configuration mode

## User Guidelines

When multiple macros are applied on a single interface, the description text is a concatenation of texts from a number of previously-applied macros.

You can verify your setting by entering the **show parser macro description** privileged EXEC mode command.

## Example

```
Switch(config) # interface gi2
Switch<config-if> # macro apply dup
Switch<config-if> # end
Switch(config) # interface gi3
Switch<config-if> # macro apply duplex $DUPLEX full $SPEED 100
Switch<config-if> # end
Switch # show parser macro description
Interface  Macro Description
-----  -----
gi2      dup
gi3      duplex
-----

Switch(config) # interface gi2
Switch<config-if> # no macro description
Switch<config-if> # end
Switch # show parser macro description
Interface  Macro Description
-----  -----
gi3      duplex
```

### 3.4 macro global

Use the **macro global** Global Configuration command to apply a macro to a switch or to apply and trace a macro configuration on a switch.

#### Syntax

```
macro global {apply | trace} macro-name [parameter-name1 {value}]  
[parameter-name2 {value}] [parameter-name3 {value}]
```

#### Parameters

- **apply**—Apply a macro to the switch.
- **trace**—Apply and trace a macro to the switch.
- **macro-name**—Specify the name of the macro.
- **parameter-name value**—(Optional) Specify unique parameter values that are specific to the switch. You can enter up to three parameter-value pairs. Parameter keyword matching is case sensitive. All matching occurrences of the parameters are replaced with the corresponding value.

#### Default Configuration

The command has no default setting.

#### Command Mode

Global Configuration mode

#### User Guidelines

You can use the **macro global trace macro-name** Global Configuration mode command to apply and show the macros running on the switch or to debug the macro in order to locate any syntax or configuration errors.

If a command fails because of a syntax error or a configuration error when you apply a macro, the macro continues to apply the remaining commands to the switch.

Use the parameter-name, value keywords to designate values specific to the switch when creating a macro that requires the assignment of unique value.

Keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value. Any full match of a keyword, even if it is part of a large string, is considered a match and replaced by the corresponding value.



If you apply a macro that contains keywords in its commands, the command fails if you do not specify the proper values for the keywords when you apply the macro. You can use the **macro global apply macro-name** with a '?' to display the help string for the macro keywords. You define the keywords in the help string using the preprocessor command **#macro keywords** when you define a macro.

When you apply a macro in Global Configuration mode, the switch automatically generates a global macro description command with the macro name. As a result, the macro name is appended to the global macro history. You can display the global macro history using the **show parser macro description** command.

### Example.

The following is an example of a macro being defined and then applied to the switch with the trace option.

```
Switch(config)# macro name console-timeout
Enter macro commands one per line. End with the character '@'.
line console
exec-timeout $timeout-interval
@
Switch(config)# macro global trace console-timeout $timeout-interval 100
Applying command... 'line console'
Applying command... 'exec-timeout 100'
Switch(config)#
```

## 3.5 macro global description

Use the **macro global description** Global Configuration command to enter a description which is used to indicate which macros have been applied to the switch. Use the **no** form of this command to remove the description.

### Syntax

**macro global description** *text*

**no macro global description**

### Parameters

**text**—Description text. The text can contain up to 160 characters.

### Default Configuration

The command has no default setting.

### Command Mode

Global Configuration mode

### User Guidelines

When multiple global macros are applied to a switch, the global description text is a concatenation of texts from a number of previously applied macros.

You can verify your settings by entering the **show parser macro description** privileged EXEC mode command.

### Examples

```
Switch(conf)# macro global description "set console timeout interval"
```

---

## 3.6 show parser macro

Use the **show parser macro** User EXEC mode command to display the parameters for all configured macros or for one macro on the switch.

### Syntax

```
show parser macro [brief / description [interface interface-id] / name macro-name]
```

### Parameters

- **brief**—Display the name of all macros.
- **description** [**interface** *interface-id*]—Display the macro descriptions for all interfaces or if an interface is specified, display the macro descriptions for that interface.
- **name** *macro-name*—Display information about a single macro identified by the macro name.

### Command Mode

User EXEC mode

## Examples

**Example 1** - This is a partial output example from the **show parser macro** command.

---

```
Switch# show parser macro

Total number of macros = 6

-----

Macro name : cisco-global
Macro type : default global
# Enable dynamic port error recovery for link state
# failures
<output truncated>

-----

Macro name : cisco-desktop
Macro type : default interface
# macro keywords $AVID
# Basic interface - Enable data VLAN only
# Recommended value for access vlan (AVID) should not be 1
switchport access vlan $AVID
switchport mode access
<output truncated>

-----description [interface
. . .
```

---

**Example 2** - This is an example of output from the **show parser macro name** command.

---

```
Switch# show parser macro standard-switch10

Macro name : standard-switch10
Macro type : customizable
macro description standard-switch10
```

---

```
# Trust QoS settings on VOIP packets
auto qos voip trust

# Allow port channels to be automatically formed
channel-protocol pagp
```

---

**Example 3** - This is an example of output from the **show parser macro brief** command.

---

```
Switch# show parser macro brief
default global : cisco-global
default interface: cisco-desktop
default interface: cisco-phone
default interface: cisco-switch
default interface: cisco-router
customizable : snmp
```

---

**Example 4** - This is an example of output from the **show parser macro description** command.

```
Switch# show parser macro description
Global Macro(s): cisco-global
```

This is an example of output from the **show parser macro description interface** command.

---

```
Switch# show parser macro description interface gi2
Interface Macro Description
-----
gi2 this is test macro
-----
```

---

## 4 RSA and Certificate Commands

---

### 4.1 `crypto key generate dsa`

The `crypto key generate dsa` Global Configuration mode command generates DSA key pairs.

#### Syntax

`crypto key generate dsa`

#### Default Configuration

DSA key pairs do not exist.

#### Command Mode

Global Configuration mode

#### User Guidelines

DSA keys are generated in pairs - one public DSA key and one private DSA key.

If the device already has DSA keys, a warning is displayed with a prompt to replace the existing keys with new keys.

This command is not saved in the router configuration. However, the keys generated by this command are saved in the private configuration (which is never displayed to the user or backed up to another device).

#### Example

The following example generates DSA key pairs.

---

```
Console(config)# crypto key generate dsa
```

---

### 4.2 `crypto key generate rsa`

The `crypto key generate rsa` Global Configuration mode command generates RSA key pairs.

#### Syntax

`crypto key generate rsa`

### Default Configuration

RSA key pairs do not exist.

### Command Mode

Global Configuration mode

### User Guidelines

RSA keys are generated in pairs - one public RSA key and one private RSA key.

If the device already has RSA keys, a warning is displayed with a prompt to replace the existing keys with new keys.

This command is not saved in the router configuration; however, the keys generated by this command are saved in the private configuration (which is never displayed to the user or backed up to another device).

### Example

The following example generates RSA key pairs.

---

```
Console(config)# crypto key generate rsa
```

---

## 4.3 show crypto key mypubkey

The **show crypto key mypubkey** Privileged EXEC mode command displays the device SSH public keys.

### Syntax

```
show crypto key mypubkey [rsa | dsa]
```

### Parameters

- **rsa**—Displays the RSA key.
- **dsa**—Displays the DSA key.

### Command Mode

Privileged EXEC mode

### Example

The following example displays the SSH public RSA keys on the device.

```
Console# show crypto key mypubkey rsa

RSA key data:

005C300D 06092A86 4886F70D 01010105 00034B00 30480241 00C5E23B 55D6AB22
04AEF1BA A54028A6 9ACC01C5 129D99E4 64CAB820 847EDAD9 DF0B4E4C 73A05DD2
BD62A8A9 FA603DD2 E2A8A6F8 98F76E28 D58AD221 B583D7A4 71020301 87685768

Fingerprint(Hex): 77:C7:19:85:98:19:27:96:C9:CC:83:C5:78:89:F8:86

Fingerprint(Bubble Babble): yteriuwt jgkljhgk yewiury hdskjfryt gfhkjgk
```

## 4.4 crypto certificate generate

The **crypto certificate generate** Global Configuration mode command generates a self-signed certificate for HTTPS.

### Syntax

```
crypto certificate number generate [key-generate [length]] [passphrase string] [cn
common-name] [ou organization-unit] [or organization] [loc location] [st state] [cu
country] [duration days]
```

### Parameters

- **number**—Specifies the certificate number. (Range: 1–2)
- **key-generate**—Regenerates SSL RSA key.
- **length**—Specifies the SSL's RSA key length. (Range: 512–2048)
- **passphrase string**—Specifies the passphrase used for exporting the certificate in PKCS12 file format. (Length: 8–96 characters)
- **cn common-name**—Specifies the fully qualified device URL or IP address. (Length: 1–64 characters)
- **ou organization-unit**—Specifies the organization-unit or department name. (Length: 1–64 characters)
- **or organization**—Specifies the organization name. (Length: 1–64 characters)

- **loc location**—Specifies the location or city name. (Length: 1–64 characters)
- **st state**—Specifies the state or province name. (Length: 1–64 characters)
- **cu country**—Specifies the country name. (Length: 2 characters)
- **duration days**—Specifies the number of days a certification is valid. (Range: 30–3650)

### Default Configuration

The default certificate number is 1.

The default SSL's RSA key length is 1024.

If **passphrase string** is not specified, the certificate is not exportable.

If **cn common-name** is not specified, it defaults to the device's lowest static IPv6 address (when the certificate is generated), or to the device's lowest static IPv4 address if there is no static IPv6 address, or to 0.0.0.0 if there is no static IP address.

If **duration days** is not specified, it defaults to 365 days.

### Command Mode

Global Configuration mode

### User Guidelines

This command is not saved in the router configuration. However, the certificate and keys generated by this command are saved in the private configuration (which is never displayed to the user or backed up to another device).

When exporting a RSA key pair to a PKCS#12 file, the RSA key pair is as secure as the passphrase. Keep the passphrase secure.

If the RSA key does not exist, you must use the parameter **key-generate**.

### Example

The following example generates a self-signed certificate for HTTPS.

---

```
Console# crypto certificate generate key-generate
```



## 4.5 crypto certificate request

The **crypto certificate request** Privileged EXEC mode command generates and displays a certificate request for HTTPS.

### Syntax

**crypto certificate** *number* **request** *common-name* [*ou organization-unit*] [*or organization*] [*loc location*] [*st state*] [*cu country*]

### Parameters

- **number**—Specifies the certificate number. (Range: 1–2)
- **common-name**—Specifies the device’s fully qualified URL or IP address. (Length: 1–64 characters)
- **ou organization-unit**—Specifies the organization-unit or department name. (Length: 1–64 characters)
- **or organization**—Specifies the organization name. (Length: 1–64 characters)
- **loc location**—Specifies the location or city name. (Length: 1–64 characters)
- **st state**—Specifies the state or province name. (Length: 1–64 characters)
- **cu country**—Specifies the country name. (Length: 2 characters)

### Command Mode

Privileged EXEC mode

### User Guidelines

Use this command to export a certificate request to a Certification Authority. The certificate request is generated in Base64-encoded X.509 format.

Before generating a certificate request, first generate a self-signed certificate using the **crypto certificate generate** Global Configuration mode command to generate the keys. The certificate fields must be re-entered.

After receiving the certificate from the Certification Authority, use the **crypto certificate import** Global Configuration mode command to import the certificate into the device. This certificate replaces the self-signed certificate.

### Example

The following example displays the certificate request for HTTPS.

```

Console# crypto certificate 1 request
-----BEGIN CERTIFICATE REQUEST-----

MIWtCCASoCAQAwYjELMAkGA1UEBhMCUFaxCzAJBgNVBAGTAkNDMQswCQYDVQQH
EwRDEMMAoGA1UEChMDZGxkMQwwCgYDVQQLEwNkbGQxCzAJBgNVBAMTAmxkMRAw
DgKOZlIhvcNAQkBFgFsmIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC8ecwQ
HdML0831i0fh/F0MV/Kib6Sz5p+3nUUenbfHp/igVPmFM+lnbqTDekb2ymCu6K
aKvEbVLF9F2LmM7VPjDBb9bb4jnxkvwW/wzDLvW2rsy5NPmH1QVl+8Ubx3GyCm
/oW93BSOFwxwEsP58kf+sPYPy+/8wwmoNtDwIDAQABoB8wHQYJKoZIhvcNAQkH
MRDjEyMwgICCAgICAICAQIMA0GCSqGSIb3DQEBBAUAA4GBAGb8UgIx7rB05m+2
m5ZZPhIwl8ARSPXwhVdJexFjbnmvcacqjPG8pIiRV6LkxryGF2bVU3jKEipcZa
g+uNpyTkDt3ZVU72pjz/fa8TF0n3

-----END CERTIFICATE REQUEST-----

CN= router.gm.com
O= General Motors
C= US

```

## 4.6 crypto certificate import

The **crypto certificate import** Global Configuration mode command imports a certificate signed by a Certification Authority for HTTPS.

### Syntax

**crypto certificate *number* import**

### Parameters

**number**—Specifies the certificate number. (Range: 1–2)

## Command Mode

Global Configuration mode

## User Guidelines

To end the session, use a blank line.

The imported certificate must be based on a certificate request created by the **crypto certificate request** privileged EXEC command.

If the public key found in the certificate does not match the device's SSL RSA key, the command fails.

This command is not saved in the router configuration. However, the certificate imported by this command is saved in the private configuration (which is never displayed to the user or backed up to another device).

## Example

The following example imports a certificate signed by Certification Authority for HTTPS.

```
Console(config)# crypto certificate 1 import
-----BEGIN CERTIFICATE-----
dHmUgUm9vdCBDZXJ0aWZpZXIwXDANBgkqhkiG9w0BAQEFAANLADBIAAkEAp4HS
nnH/xQSGA2ffkRBwU2XIxb7n8VPsTmlxyJlt11a1GaqchfMqqe0kmfhcoHSWr
yf1FpD0MWOTgDAwIDAQABo4IBojCCA4wEwYJKwYBBAGCNxQCBAYeBABDAEEw
CwR0PBAQDAgFGMA8GA1UdEwEB/wQFMAMBAf8wHQYDVR0OBByEFaf4MT9BRD47
ZvKBAEL9Ggp+6MIIBNgYDVR0fBIIBLTCCASKwgdKggc+ggcyGgclsZGFwOi8v
L0VByb3h5JTIwU29mdHdhcmU1MjBSb290JTIwQ2VydG1maWVyLENOPXN1cnZl
-----END CERTIFICATE-----

Certificate imported successfully.

Issued to: router.gm.com

Issued by: www.verisign.com

Valid from: 8/9/2003 to 8/9/2004

Subject: CN= router.gm.com, O= General Motors, C= US
```

```
Finger print: DC789788 DC88A988 127897BC BB789788
```

---

## 4.7 show crypto certificate mycertificate

The **show crypto certificate mycertificate** Privileged EXEC mode command displays the device SSL certificates.

### Syntax

**show crypto certificate mycertificate** [*number*]

### Parameters

**number**—Specifies the certificate number. (Range: 1–2)

### Command Mode

Privileged EXEC mode

### Example

The following example displays SSL certificate # 1 present on the device.

---

```
Console# show crypto certificate mycertificate 1
-----BEGIN CERTIFICATE-----
dHmUgUm9vdCBDZXJ0aWZpZXIwXDANBgkqhkiG9w0BAQEFAANLADBIAkEAp4HS
nnH/xQSGA2ffkRBwU2XIxb7n8VPsTmlxyJlt1l1a1GaqchfMqqe0kmfhcoHSWr
yf1FpD0MWOTgDAwIDAQABo4IBojCCAZ4wEwYJKwYBBAGCNxQCBAYeBABDAEEw
CwR0PBAQDAgFGMA8GA1UdEwEB/wQFMAMBAf8wHQYDVR0OBBYEFaf4MT9BRD47
ZvKBAEL9Ggp+6MIIBNgYDVR0fBIIBLTCCASkwdKggc+ggcyGgclsZGFwOi8v
L0VByb3h5JTIwU29mdHdhcmU1mjBSb290JTIwQ2Vydg1maWVyLENOPXN1cnZl
-----END CERTIFICATE-----

Issued by: www.verisign.com
Valid from: 8/9/2003 to 8/9/2004
Subject: CN= router.gm.com, O= General Motors, C= US
```

---

Finger print: DC789788 DC88A988 127897BC BB789788

## 5 System Management Commands

### 5.1 ping

Use the **ping** EXEC mode command to send ICMP echo request packets to another node on the network.

#### Syntax

```
ping [ip] {ipv4-address / hostname} [size packet_size] [count packet_count] [timeout time_out]
```

```
ping ipv6 {ipv6-address / hostname} [size packet_size] [count packet_count] [timeout time_out]
```

#### Parameters

- **ip**—Use IPv4 to check the network connectivity.
- **ipv6**—Use IPv6 to check the network connectivity.
- **ipv4-address**—IPv4 address to ping.
- **ipv6-address**—Unicast or Multicast IPv6 address to ping. When the IPv6 address is a Link Local address (IPv6Z address), the outgoing interface name must be specified. Refer to the User Guidelines of this command for the interface name syntax.
- **hostname**—Hostname to ping (160 characters. Maximum label size: 63.)
- **size packet\_size**—Number of bytes in the packet not including the VLAN tag. The default is 64 bytes. (IPv4:64–1518, IPv6: 68–1518)
- **count packet\_count**—Number of packets to send, from 1 to 65535 packets. The default is 4 packets. If 0 is entered, it pings until stopped (0–65535).
- **time time-out**—Timeout in milliseconds to wait for each reply, from 50 to 65535 milliseconds. The default is 2000 milliseconds (50–65535).

#### Default Usage

N/A

#### Command Mode

EXEC mode

## User Guidelines

Press **Esc** to stop pinging. Following are sample results of the ping command:

- **Destination does not respond**—If the host does not respond, a “no answer from host” appears within 10 seconds.
- **Destination unreachable**—The gateway for this destination indicates that the destination is unreachable.
- **Network or host unreachable**—The switch found no corresponding entry in the route table.

The format of an **IPv6Z** address is: *<ipv6-link-local-address>%<interface-name>*

- **interface-name** = *vlan<integer> | po<integer> | isatap<integer> | <physical-port-name> | 0*
- **integer** = *<decimal-number> | <integer><decimal-number>*
- **decimal-number** = *0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9*
- **physical-port-name** = Designated port number, for example gi1. If the physical-port-name = 0, then it is not defined and the default interface is used.

When using the ping **ipv6** command to check network connectivity of a directly attached host using its link local address, the egress interface may be specified in the **IPv6Z** format. If the egress interface is not specified, the default interface is selected.

When using the ping **ipv6** command with a Multicast address, the information displayed is taken from all received echo responses.

## Examples

### Example 1- Ping an IP address.

---

```
Console> ping ip 10.1.1.1

Pinging 10.1.1.1 with 64 bytes of data:

64 bytes from 10.1.1.1: icmp_seq=0. time=11 ms
64 bytes from 10.1.1.1: icmp_seq=1. time=8 ms
64 bytes from 10.1.1.1: icmp_seq=2. time=8 ms
64 bytes from 10.1.1.1: icmp_seq=3. time=7 ms
```

```
----10.1.1.1 PING Statistics----  
4 packets transmitted, 4 packets received, 0% packet loss  
round-trip (ms) min/avg/max = 7/8/11
```

---

### Example 2 - Ping a site.

```
Console> ping ip yahoo.com  
  
Pinging yahoo.com [66.218.71.198] with 64 bytes of data:  
64 bytes from 10.1.1.1: icmp_seq=0. time=11 ms  
64 bytes from 10.1.1.1: icmp_seq=1. time=8 ms  
64 bytes from 10.1.1.1: icmp_seq=2. time=8 ms  
64 bytes from 10.1.1.1: icmp_seq=3. time=7 ms  
  
----10.1.1.1 PING Statistics----  
4 packets transmitted, 4 packets received, 0% packet loss  
round-trip (ms) min/avg/max = 7/8/11
```

---

### Example 3 - Ping an IPv6 address.

```
console> ping ipv6 3003::11  
  
Pinging 3003::11 with 64 bytes of data:  
64 bytes from 3003::11: icmp_seq=1. time=0 ms  
64 bytes from 3003::11: icmp_seq=2. time=50 ms  
64 bytes from 3003::11: icmp_seq=3. time=0 ms  
64 bytes from 3003::11: icmp_seq=4. time=0 ms  
  
----3003::11 PING Statistics----  
4 packets transmitted, 4 packets received, 0% packet loss  
round-trip (ms) min/avg/max = 0/12/50
```

---

```
console> ping ipv6 FF02::1  
  
Pinging FF02::1 with 64 bytes of data:  
64 bytes from 3003::11: icmp_seq=1. time=0 ms
```



```
64 bytes from 3003::33: icmp_seq=1. time=70 ms
64 bytes from 3003::11: icmp_seq=2. time=0 ms
64 bytes from 3003::55: icmp_seq=1. time=1050 ms
64 bytes from 3003::33: icmp_seq=2. time=70 ms
64 bytes from 3003::55: icmp_seq=2. time=1050 ms
64 bytes from 3003::11: icmp_seq=3. time=0 ms
64 bytes from 3003::33: icmp_seq=3. time=70 ms
64 bytes from 3003::11: icmp_seq=4. time=0 ms
64 bytes from 3003::55: icmp_seq=3. time=1050 ms
64 bytes from 3003::33: icmp_seq=4. time=70 ms
64 bytes from 3003::55: icmp_sq=4. time=1050 ms

---- FF02::1 PING Statistics----
4 packets transmitted, 12 packets received
```

---

## 5.2 traceroute

To display the routes that packets will take when traveling to their destination, use the **traceroute** EXEC mode command.

### Syntax

```
traceroute ip {ipv4-address / hostname} [size packet_size] [ttl max-ttl] [count packet_count] [timeout time_out] [source ip-address] [tos tos]
```

```
traceroute ipv6 {ipv6-address / hostname} [size packet_size] [ttl max-ttl] [count packet_count] [timeout time_out] [source ip-address] [tos tos]
```

### Parameters

- **ip**—Use IPv4 to discover the route.
- **ipv6**—Use IPv6 to discover the route.
- **ipv4-address**—IPv4 address of the destination host. (Range: Valid IP address)
- **ipv6-address**—IPv6 address of the destination host.

- **hostname**—Hostname of the destination host. (Range: 1–160 characters. Maximum label size: 63)
- **size *packet\_size***—Number of bytes in the packet not including the VLAN tag. The default is 64 bytes. (IPv4:64-1518, IPv6: 68-1518)
- **ttl *max-ttl***—The largest TTL value that can be used. The default is 30. The **traceroute** command terminates when the destination is reached or when this value is reached. (Range: 1–255)
- **count *packet\_count***—The number of probes to be sent at each TTL level. The default count is 3. (Range: 1–10)
- **timeout *time\_out***—The number of seconds to wait for a response to a probe packet. The default is 3 seconds. (Range: 1–60)
- **source *ip-address***—One of the interface addresses of the device to use as a source address for the probes. The device selects the optimal source address by default. (Range: Valid IP address)
- **tos *tos***—The Type-Of-Service byte in the IP Header of the packet. (Range: 0–255)

### Default Usage

N/A

### Command Mode

EXEC mode

### User Guidelines

The traceroute command works by taking advantage of the error messages generated by routers when a datagram exceeds its time-to-live (TTL) value.

The traceroute command starts by sending probe datagrams with a TTL value of one. This causes the first router to discard the probe datagram and send back an error message. The traceroute command sends several probes at each TTL level and displays the round-trip time for each.

The traceroute command sends out one probe at a time. Each outgoing packet can result in one or two error messages. A "time exceeded" error message indicates that an intermediate router has seen and discarded the probe. A "destination unreachable" error message indicates that the destination node has received the probe and discarded it because it could not deliver the packet. If the timer goes off before a response comes in, the traceroute command prints an asterisk (\*).

The traceroute command terminates when the destination responds, when the maximum TTL is exceeded, or when the user interrupts the trace with Esc.

The traceroute command is not relevant to IPv6 link local addresses.

### Example

```
Router> traceroute ip umaxpl.physics.lsa.umich.edu

Type Esc to abort.

Tracing the route to umaxpl.physics.lsa.umich.edu (141.211.101.64)

 0  i2-gateway.stanford.edu (192.68.191.83)  0 msec 0 msec 0 msec
 1  STAN.POS.calren2.NET (171.64.1.213)  0 msec 0 msec 0 msec
 2  SUNV--STAN.POS.calren2.net (198.32.249.73)  1 msec 1 msec 1 msec
 3  Abilene--QSV.POS.calren2.net (198.32.249.162)  1 msec 1 msec 1 msec
 4  kscyng-snvang.abilene.ucaid.edu (198.32.8.103)  33 msec 35 msec 35 msec
 5  iplsng-kscyng.abilene.ucaid.edu (198.32.8.80)  47 msec 45 msec 45 msec
 6  so-0-2-0x1.aal.mich.net (192.122.183.9)  56 msec 53 msec 54 msec
 7  atm1-0x24.michnet8.mich.net (198.108.23.82)  56 msec 56 msec 57 msec
 8  * * *
 9  A-ARB3-LSA-NG.c-SEB.umnet.umich.edu(141.211.5.22) 58 msec 58msec 58 msec
10  umaxpl.physics.lsa.umich.edu (141.211.101.64)  62 msec 63 msec 63 msec

Trace completed
```

The following table describes the significant fields shown in the display:

Field	Description
1	Indicates the sequence number of the router in the path to the host.
i2-gateway.stanford.edu	Host name of this router.
192.68.191.83	IP address of this router.
1 msec 1 msec 1 msec	Round-trip time for each of the probes that are sent.

The following are characters that can appear in the traceroute command output:

Field	Description
*	The probe timed out.
?	Unknown packet type.
A	Administratively unreachable. Usually, this output indicates that an access list is blocking traffic.
F	Fragmentation required and DF is set.
H	Host unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.
R	Fragment reassembly time exceeded
S	Source route failed.
U	Port unreachable.

### 5.3 telnet

The **telnet** EXEC mode command enables logging on to a host that supports Telnet.

#### Syntax

```
telnet {ip-address | hostname} [port] [keyword...]
```

#### Parameters

- **ip-address**—Specifies the destination host IP address (IPv4 or IPv6).
- **hostname**—Specifies the destination host name. (Length: 1-160 characters. Maximum label length: 63 characters.)
- **port**—Specifies the decimal TCP port number or one of the keywords listed in the Ports table in the User Guidelines.

- **keyword**—Specifies the one or more keywords listed in the Keywords table in the User Guidelines.

### Default Configuration

The default port is the Telnet port (23) on the host.

By default, Telnet is disabled.

### Command Mode

EXEC mode

### User Guidelines

Telnet software supports special Telnet commands in the form of Telnet sequences that map generic terminal control functions to operating system-specific functions. To enter a Telnet sequence, press the escape sequence keys (Ctrl-shift-6) followed by a Telnet command character.

### Special Telnet Sequences

Telnet Sequence	Purpose
Ctrl-shift-6-b	Break
Ctrl-shift-6-c	Interrupt Process (IP)
Ctrl-shift-6-h	Erase Character (EC)
Ctrl-shift-6-o	Abort Output (AO)
Ctrl-shift-6-t	Are You There? (AYT)
Ctrl-shift-6-u	Erase Line (EL)

At any time during an active Telnet session, available Telnet commands can be listed by pressing the `?/help` keys at the system prompt.

A sample of this list follows.

---

```

Console> ?/help
[Special telnet escape help]
^^ B sends telnet BREAK
^^ C sends telnet IP

```

```

^^ H sends telnet EC
^^ O sends telnet AO
^^ T sends telnet AYT
^^ U sends telnet EL

?/help suspends the session (return to system command prompt)

```

Several concurrent Telnet sessions can be opened, enabling switching between the sessions. To open a subsequent session, the current connection has to be suspended by pressing the escape sequence keys (Ctrl-shift-6) and x to return to the system command prompt. Then open a new connection with the telnet EXEC mode command.

This command lists concurrent Telnet connections to remote hosts that were opened by the current Telnet session to the local device. It does not list Telnet connections to remote hosts that were opened by other Telnet sessions.

### Keywords Table

Options	Description
<b>/echo</b>	Enables local echo.
<b>/quiet</b>	Prevents onscreen display of all messages from the software.
<b>/source-interface</b>	Specifies the source interface.
<b>/stream</b>	Turns on stream processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process Telnet options and can be appropriate for connections to ports running UNIX-to-UNIX Copy Program (UUCP) and other non-Telnet protocols.
<b>Ctrl-shift-6 x</b>	Returns to the System Command Prompt.

### Ports Table

Keyword	Description	Port Number
BGP	Border Gateway Protocol	179
chargen	Character generator	19

Keyword	Description	Port Number
cmd	Remote commands	514
daytime	Daytime	13
discard	Discard	9
domain	Domain Name Service	53
echo	Echo	7
exec	Exec	512
finger	Finger	79
ftp	File Transfer Protocol	21
ftp-data	FTP data connections	20
gopher	Gopher	70
hostname	NIC hostname server	101
ident	Ident Protocol	113
irc	Internet Relay Chat	194
klogin	Kerberos login	543
kshell	Kerberos shell	544
login	Login	513
lpd	Printer service	515
nntp	Network News Transport Protocol	119
pim-auto-rp	PIM Auto-RP	496
pop2	Post Office Protocol v2	109
pop3	Post Office Protocol v3	110
smtp	Simple Mail Transport Protocol	25
sunrpc	Sun Remote Procedure Call	111

---

Keyword	Description	Port Number
syslog	Syslog	514
tacacs	TAC Access Control System	49
talk	Talk	517
telnet	Telnet	23
time	Time	37
uucp	Unix-to-Unix Copy Program	540
whois	Nickname	43
www	World Wide Web	80

### Example

The following example displays logging in to IP address 176.213.10.50 via Telnet.

---

```
Console> telnet 176.213.10.50
```

---

## 5.4 resume

The **resume** EXEC mode command enables switching to another open Telnet session.

### Syntax

```
resume [connection]
```

### Parameters

**connection**—Specifies the connection number. (Range: 1-4 connections.)

### Default Configuration

The default connection number is that of the most recent connection.

### Command Mode

EXEC mode



### Example

The following command switches to open Telnet session number 1.

---

```
Console> resume 1
```

---

## 5.5 hostname

The **hostname** Global Configuration mode command specifies or modifies the device host name. Use the **no** form of the command to remove the existing host name.

### Syntax

**hostname** *name*

**no hostname**

### Parameters

**Name**—Specifies the device host name. (Length: 1-63). The hostname must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, and hyphens.

### Default Configuration

No host name is defined.

### Command Mode

Global Configuration mode

### Example

The following example specifies the device host name as 'enterprise'.

---

```
Console(config)# hostname enterprise
enterprise(config)#
```

---

## 5.6 reload

The **reload** Privileged EXEC mode command reloads the operating system.

**Syntax****reload****Parameters**

N/A

**Default Usage**

N/A

**Command Mode**

Privileged EXEC mode

**Example**

The following example reloads the operating system.

---

```
Console# reload
```

```
This command will reset the whole system and disconnect your current session.
```

```
Do you want to continue? (y/n) [n]
```

---

## 5.7 **service cpu-utilization**

The **service cpu-utilization** Global Configuration mode command enables measuring CPU utilization. Use the **no** form of this command to restore the default configuration.

**Syntax****service cpu-utilization****no service cpu-utilization****Parameters**

N/A

**Default Configuration**

Measuring CPU utilization is disabled.

**Command Mode**

Global Configuration mode

**User Guidelines**

Use the **service cpu utilization** command to measure information on CPU utilization.

**Example**

The following example enables measuring CPU utilization.

---

```
Console(config)# service cpu-utilization
```

---

**5.8 show cpu utilization**

The **show cpu utilization** Privileged EXEC mode command displays information about CPU utilization.

**Syntax**

**show cpu utilization**

**Parameters**

N/A

**Default Usage**

N/A

**Command Mode**

Privileged EXEC mode

**User Guidelines**

Use the **show cpu-utilization** command to enable measuring CPU utilization.

**Example**

The following example displays CPU utilization information.

---

```
Console# show cpu utilization
CPU utilization service is on.
CPU utilization
-----
five seconds: 5%; one minute: 3%; five minutes: 3%
```

---

## 5.9 show users

The **show users** EXEC mode command displays information about the active users.

### Syntax

**show users**

### Parameters

N/A

### Default Usage

N/A

### Command Mode

EXEC mode

### Example

The following example displays information about the active users.

---

```
Console# show users
```

Username	Protocol	Location
-----	-----	-----
Bob	Serial	
John	SSH	172.16.0.1
Robert	HTTP	172.16.0.8
Betty	Telnet	172.16.1.7
Sam		172.16.1.6

---

## 5.10 show sessions

The **show sessions** EXEC mode command displays open Telnet sessions.

### Syntax

**show sessions**

### Parameters

N/A

### Default Usage

N/A

### Command Mode

EXEC mode

### User Guidelines

The **show sessions** command displays Telnet sessions to remote hosts opened by the current Telnet session to the local device. It does not display Telnet sessions to remote hosts opened by other Telnet sessions to the local device.

**Example**

The following example displays open Telnet sessions.

---

```

Console# show sessions

Connection  Host                Address            Port    Byte
-----
1           Remote router      172.16.1.1        23     89
2           172.16.1.2         172.16.1.2        23     8

```

The following table describes significant fields shown above.

Field	Description
Connection	The connection number.
Host	The remote host to which the device is connected through a Telnet session.
Address	The remote host IP address.
Port	The Telnet TCP port number.
Byte	The number of unread bytes for the user to see on the connection.

**5.11 show system**

The **show system** EXEC mode command displays system information.

**Syntax**

**show system**

**Parameters**

There are no parameters for this command.

**Command Mode**

EXEC mode

### Example

The following example displays the system information.

---

```
console# show system

switch151400(config)#exit

switch151400#show system

System Description:                20-port Gigabit Managed Switch
System Up Time (days,hour:min:sec): 03,02:27:46
System Contact:
System Name:                        switch151400
System Location:
System MAC Address:                 00:24:ab:15:14:00
System Object ID:                   1.3.6.1.4.1.9.6.1.83.20.1
```

---

## 5.12 show version

The **show version** EXEC mode command displays system version information.

### Syntax

**show version**

### Parameters

N/A

### Default Usage

N/A

### Command Mode

EXEC mode

### Example

The following example displays system version information.

---

```
console > show version

SW Version      1.1.0.5 ( date 15-Sep-2010 time 10:31:33 )
Boot Version    1.1.0.2 ( date 04-Sep-2010 time 21:51:53 )
HW Version      V01
```

---

### 5.13 show version md5

Use the **show version md5** EXEC mode command to display external MD5 digest of firmware.

#### Syntax

**show version md5**

#### Parameters

N/A

#### Default Usage

N/A

#### Command Mode

EXEC mode

#### Example

---

```
>show version md5

Filename      Status      MD5 Digest
-----
image1        Active      23FA000012857D8855AABC7577AB5562
image2        Not Active  23FA000012857D8855AABEA7451265456
boot          Active      23FA000012857D8855AABC7577AB8999
mage1         Not Active  23FA000012857D8855AABC757FE693844
image2        Active      23FA000012857D8855AABC7577AB5562
boot          Active      23FA000012857D8855AABC7577AC9999
```



---

## 5.14 system resources routing

The **system resources routing** Global Configuration mode command configures the routing table maximum size. Use the **no** form of this command to return to the default size.

### Syntax

**system resources routing** *routes hosts interfaces*

**no system resources routing**

### Parameters

- **routes**—Specifies the maximum number of remote networks in the routing table.
- **hosts**—Specifies the maximum number of directly attached hosts.
- **interfaces**—Specifies the maximum number of IP interfaces.

### Default Configuration

Hosts: 2-100, default = 100

Routes: 1-32, default = 32

IP Interfaces: 2-32, default = 32

### Command Mode

Global Configuration mode

### User Guidelines

The settings are effective after reboot.

### Example

The following example configures the routing table maximum size.

---

```
Console# system resources routing 20 23 5
```

## 5.15 show system resources

The **show system resources routings** EXEC mode command displays system routing and tcam resource information.

### Syntax

```
show system resources {routing| tcam}
```

### Parameters

**routing**—Displays the number of hosts, routers and IP interfaces that are available.

**tcam**—Displays the number of TCAM rules that are available.

### Command Mode

EXEC mode

### Example

**Example 1** - The following example displays the system routing resources information. The values in the Current Value column show what resources are currently available. The values in the After Reboot Value column show what resources will be available after reboot as a result of system resources routing command.

```
Console> show system resources routing
```

Parameters	Current Value	After Reboot Value
Hosts:	100	100
Routes:	32	32
IP Interfaces:	32	32

**Example 2** - The following example displays the system routing resources information. The values in the Current Value column show what resources are currently available. The values in the After Reboot Value column show what resources will be available after reboot as a result of system resources routing command.

```
Console> show system resources tcam
```

TCAM resources

-----

Maximum number of miscellaneous TCAM rules:	314
Used number of miscellaneous TCAM rules:	26
Maximum number of routing TCAM rules:	196
Used number of routing TCAM rules:	3

---

## 5.16 set system mode

The **set system mode** Privileged EXEC mode command puts the device into switch mode (Layer 2 mode) or router mode (Layer 3 mode).

### Syntax

**set system mode** {*router* / *switch*}

### Parameters

- **router**—Specifies that the device functions as a switch-router.
- **switch**—Specifies that the device functions as a switch.

### Default Configuration

The default configuration is switch mode (Layer 2).

### Command Mode

Privileged EXEC mode

### User Guidelines

After executing the command, the Startup Configuration file is deleted and the device is rebooted. It is highly recommended to back up the Startup Configuration file before executing this command.

### Example

The following example configures the device to function as a switch-router (Layer 3), with QoS and Policy based VLANs.

---

```
Console# set system mode router
```

---

## 5.17 show system mode

The **show system mode** EXEC mode command displays information on features control.

### Syntax

**show system mode**

### Parameters

N/A

### Default Usage

N/A

### Command Mode

EXEC mode

### Example

The following example displays system mode information.

---

```
Console> show system mode
Feature                               State
-----                               -
Mode:                                 Router
Qos:                                   Active
Policy-based-vlans:                   Active
```

---

## 5.18 show system languages

The **show system languages** EXEC mode command displays the list of supported languages.

### Syntax

**show system languages**

**Parameters**

N/A

**Default Usage**

N/A

**Command Mode**

EXEC mode

**Example**

The following example displays the languages configured on the device. Number of Sections indicates the number of languages permitted on the device.

---

```
Console> show system languages
```

Language Name	Unicode Name	Code	Num of Sections
English	English	en-US	2
Japanese	日本語	ja-JP	2

**5.19 show system tcam utilization**

The **show system tcam utilization** EXEC mode command displays the Ternary Content Addressable Memory (TCAM) utilization.

**Syntax****show system tcam utilization****Parameters**

N/A

**Default Usage**

N/A

**Command Mode**

EXEC mode

**Example**

The following example displays TCAM utilization information.

---

```
Console> show system tcam utilization
TCAM utilization: 58%
```

**5.20 show services tcp-udp**

Use the **show services tcp-udp** Privileged EXEC mode command to display information about the active TCP and UDP services.

**Syntax**

**show services tcp-udp**

**Parameters**

N/A

**Command Mode**

Privileged EXEC mode

**User Guidelines**

The output does not show sessions where the device is a TCP/UDP client.

**Examples**


---

```
Console # show services tcp-udp
```

Type	Local IP Address	Remote IP address	Service Name	State
TCP	All:22		SSH	LISTEN
TCP	All:23		Telnet	LISTEN
TCP	All:80		HTTP	LISTEN
TCP	All:443		HTTPS	LISTEN
TCP	172.16.1.1:23	172.16.1.18:8789	Telnet	ESTABLISHED
TCP6	All-23		Telnet	LISTEN

---

```
TCP6   fe80::200:b0ff:fe00:0-23           Telnet
       fe80::200:b0ff:fe00:0-8999           ESTABLISHED
UDP    All:161                             SNMP
UDP6A  11-161                               SNMP
```

---

## 5.21 show system id

The **show system id** EXEC mode command displays the system identity information.

### Syntax

**show system id**

### Parameters

There are no parameters for this command.

### Command Mode

EXEC mode

### Example

The following example displays the system identity information.

---

```
Console> show system id
serial number 114
```

---

## 5.22 show cpu input rate

The **show cpu input rate** EXEC mode command displays the rate of input frames to the CPU in packets per seconds (pps).

### Syntax

**show cpu input rate**

### Command Mode

EXEC mode

---

## User Guidelines

### Example

The following example displays CPU input rate information.

---

```
Console# show cpu input rate
Input Rate to CPU is 1030 pps.
```



---

## 6 Clock Commands

---

### 6.1 clock set

The **clock set** Privileged EXEC mode command manually sets the system clock.

#### Syntax

**clock set** *hh:mm:ss* {[*day month*] | [*month day*]} *year*

#### Parameters

- **hh:mm:ss**—Specifies the current time in hours (military format), minutes, and seconds. (Range: hh: 0-23, mm: 0-59, ss: 0-59)
- **day**—Specifies the current day of the month. (Range: 1-31)
- **month**—Specifies the current month using the first three letters of the month name. (Range: Jan–Dec)
- **year**—Specifies the current year. (Range: 2000–2037)

#### Command Mode

Privileged EXEC mode

#### User Guidelines

It is recommended that the user enter the local clock time and date.

#### Example

The following example sets the system time to 13:32:00 on March 7th, 2005.

---

```
Console# clock set 13:32:00 7 Mar 2005
```

---

### 6.2 clock source

The **clock source** Global Configuration mode command configures an external time source for the system clock. Use the **no** form of this command to disable the external time source.

### Syntax

**clock source {sntp}**

**no clock source**

### Parameters

**sntp**—Specifies that an SNTP server is the external clock source.

### Default Configuration

There is no external clock source.

### Command Mode

Global Configuration mode

### Example

The following example configures an SNTP server as an external time source for the system clock.

---

```
Console(config)# clock source sntp
```

---

## 6.3 clock timezone

Use the **clock timezone** Global Configuration command to set the time zone for display purposes. Use the **no** form of this command to set the time to Coordinated Universal Time (UTC) or Greenwich Mean Time (GMT), which is the same.

### Syntax

**clock timezone** *zone hours-offset [minutes-offset]*

**no clock timezone**

### Parameters

- **zone**—The acronym of the time zone.(Range: Up to 4 characters)
- **hours-offset**—Hours difference from UTC. (Range: (-12)-(+13))
- **minutes-offset**—Minutes difference from UTC. (Range: 0-59)

### Default Configuration

Offsets are 0.

Acronym is empty.

### Command Mode

Global Configuration mode

### User Guidelines

The system internally keeps time in UTC, so this command is used only for display purposes and when the time is manually set.

### Example

```
console(config)# clock timezone abc +2 minutes 32
```

---

## 6.4 clock summer-time

Use one of the formats of the **clock summer-time** Global Configuration command to configure the system to automatically switch to summer time (Daylight Saving Time). Use the **no** form of this command to configure the software not to automatically switch to summer time.

### Syntax

**clock summer-time** *zone recurring* *{usa | eu | {week day month hh:mm week day month hh:mm}}* [*offset*]

**clock summer-time** *zone date* *day month year hh:mm date month year hh:mm* [*offset*]

**clock summer-time** *zone date* *month day year hh:mm month day year hh:mm* [*offset*]

**no clock summer-time**

### Parameters

- **zone**—The acronym of the time zone to be displayed when summer time is in effect. (Range: up to 4 characters)

- **recurring**—Indicates that summer time starts and ends on the corresponding specified days every year.
- **date**—Indicates that summer time starts on the first date listed in the command and ends on the second date in the command.
- **usa**—The summer time rules are the United States rules.
- **eu**—The summer time rules are the European Union rules.
- **week**—Week of the month. Can be 1–4, first, last.
- **day**—Day of the week (first three characters by name, such as Sun).
- **date**—Date of the month. (Range: 1–31)
- **month**—Month (first three characters by name, such as Feb).
- **year**—year (no abbreviation). (Range: 2000–2097)
- **hh:mm**—Time (military format) in hours and minutes. (Range: hh:mmhh: 0-23, mm: 0-59)
- **offset**—Number of minutes to add during summer time (default is 60). (Range: 1440)

### Default Configuration

Summer time is disabled.

### Command Mode

Global Configuration mode

### User Guidelines

In both the **date** and **recurring** forms of the command, the first part of the command specifies when summer time begins, and the second part specifies when it ends. All times are relative to the local time zone. The start time is relative to standard time. The end time is relative to summer time. If the starting month is chronologically after the ending month, the system assumes that you are in the southern hemisphere.

USA rules for Daylight Saving Time:

- From 2007:
  - Start: Second Sunday in March
  - End: First Sunday in November

- Time: 2 AM local time
- Before 2007:
  - Start: First Sunday in April
  - End: Last Sunday in October
  - Time: 2 AM local time

EU rules for Daylight Saving Time:

- Start: Last Sunday in March
- End: Last Sunday in October
- Time: 1.00 am (01:00) Greenwich Mean Time (GMT)

### Example

```
console(config)# clock summer-time abc date apr 1 2010 09:00 aug 2 2010 09:00
```

## 6.5 clock dhcp timezone

Use the **clock dhcp timezone** Global Configuration command to specify that the timezone and the Summer Time (Daylight Saving Time) of the system can be taken from the DHCP Timezone option. Use the **no** form of this command disable this option.

### Syntax

**clock dhcp timezone**

**no clock dhcp timezone**

### Parameters

N/A

### Default Configuration

Disabled

### Command Mode

Global Configuration mode

### User Guidelines

The TimeZone taken from the DHCP server has precedence over the static TimeZone. If the TimeZone does not exist in the DHCP-TimeZone option, the static configuration should be active.

The Summer Time taken from the DHCP server has precedence over static SummerTime. If the Summer Time does not exist in the DHCP-TimeZone option, the static configuration should be active.

The TimeZone and SummerTime remain effective after the IP address lease time has expired.

The TimeZone and SummerTime that are taken from the DHCP server are cleared after reboot.

When the user disables taking the TimeZone and Summer Time from the DHCP server, the dynamic Time Zone and Summer Time from the DHCP server are cleared.

In case of multiple DHCP-enabled interfaces, the last accepted DHCP-Time Zone option overrides any previous DHCP-Time Zone option. This means that the last accepted DHCP-Time Zone option overrides the previous Time Zone and the Summer Time, even if it includes only one of them.

Disabling the DHCP client from where the DHCP-TimeZone option was taken, clears the dynamic Time Zone and Summer Time configuration.

### Example

```
console(config)# clock dhcp timezone
```

## 6.6 sntp authentication-key

The **sntp authentication-key** Global Configuration mode command defines an authentication key for Simple Network Time Protocol (SNTP). Use the **no** form of this command to remove the authentication key for SNTP.

### Syntax

```
sntp authentication-key key-number md5 key-value
```

```
no sntp authentication-key key-number
```

### Parameters

- **key-number**—Specifies the key number. (Range: 1–4294967295)

- **md5 *key-value***—Specifies the key value. (Length: 1–8 characters)

### Default Configuration

No authentication key is defined.

### Command Mode

Global Configuration mode

### Examples

The following example defines the authentication key for SNTP.

---

```
Console(config)# sntp authentication-key 8 md5 ClkKey
```

---

```
Device(config)# sntp authentication-key 8 md5 ClkKey
```

```
Device(config)# sntp trusted-key 8
```

```
Device(config)# sntp authenticate
```

---

## 6.7 sntp authenticate

The **sntp authenticate** Global Configuration mode command enables authentication for received Simple Network Time Protocol (SNTP) traffic from servers. Use the **no** form of this command to disable the feature.

### Syntax

**sntp authenticate**

**no sntp authenticate**

### Parameters

N/A

### Default Configuration

Authentication is disabled.

### Command Mode

Global Configuration mode

### User Guidelines

The command is relevant for both unicast and broadcast.

### Examples

The following example enables authentication for received SNTP traffic and sets the key and encryption key.

---

```
Console(config)# sntp authenticate
```

---

```
device(config)# sntp authentication-key 8 md5 ClkKey
```

```
device(config)# sntp trusted-key 8
```

```
device(config)# sntp authenticate
```

---

## 6.8 sntp trusted-key

The **sntp trusted-key** Global Configuration mode command authenticates the identity of the system with which Simple Network Time Protocol (SNTP) synchronizes. Use the **no** form of this command to disable system identity authentication.

### Syntax

```
sntp trusted-key key-number
```

```
no sntp trusted-key key-number
```

### Parameters

**key-number**—Specifies the key number of the authentication key to be trusted. (Range: 1–4294967295)

### Default Configuration

No keys are trusted.

### Command Mode

Global Configuration mode

### User Guidelines

The command is relevant for both received unicast and broadcast.



## Examples

The following example authenticates key 8.

---

```
Console(config)# sntp trusted-key 8
```

---

```
Device(config)# sntp authentication-key 8 md5 ClkKey
```

```
Device(config)# sntp trusted-key 8
```

```
Device(config)# sntp authenticate
```

---

## 6.9 sntp broadcast client enable

The **sntp broadcast client enable** Global Configuration mode command enables Simple Network Time Protocol (SNTP) Broadcast clients. Use the no form of this command to disable SNTP Broadcast clients.

### Syntax

**sntp broadcast client enable**

**no sntp broadcast client enable**

### Default Configuration

The SNTP Broadcast client is disabled.

### Command Mode

Global Configuration mode

### User Guidelines

Use the **sntp broadcast client enable** Interface Configuration mode command to enable the SNTP Broadcast client on a specific interface.

After entering this command, you must enter [clock source](#) for the command to be run.

### Example

The following example enables SNTP Broadcast clients.

---

```
Console(config)# sntp broadcast client enable
```

---

---

## 6.10 `sntp unicast client enable`

The `sntp unicast client enable` Global Configuration mode command enables the device to use Simple Network Time Protocol (SNTP)-predefined Unicast clients. Use the `no` form of this command to disable the SNTP Unicast clients.

### Syntax

`sntp unicast client enable`

`no sntp unicast client enable`

### Parameters

N/A

### Default Configuration

The SNTP unicast client is disabled.

### Command Mode

Global Configuration mode

### User Guidelines

Use the `sntp server` Global Configuration mode command to define SNTP servers.

### Example

The following example enables the device to use SNTP Unicast clients.

---

```
Console(config)# sntp unicast client enable
```

---

## 6.11 `sntp server`

The `sntp server` Global Configuration mode command configures the device to use the Simple Network Time Protocol (SNTP) to request and accept Network Time Protocol (NTP) traffic from a specified server (meaning to accept system time from an SNTP server). Use the `no` form of this command to remove a server from the list of SNTP servers.

## Syntax

```
sntp server {ipv4-address | ipv6-address | ipv6z-address | hostname} [poll] [key  
keyid]
```

```
no sntp server {ipv4-address | ipv6-address | ipv6z-address | hostname}
```

## Parameters

- **ipv4-address**—Specifies the server IPv4 address.
- **ipv6-address**—Specifies the server IPv6 address. A Link Local address (IPv6Z address) can be defined.
- **pv6z-address**—Specifies the IPv6Z address to ping. The IPv6Z address format is: *ipv6-link-local-address*{%*interface-name*}. The subparameters are:
  - **ipv6-link-local-address**—Specifies the IPv6 Link Local address.
  - **interface-name**—Specifies the outgoing interface name. The interface name has the format: *vlan*{*integer*} | *po*{*integer*} | *isatap*{*integer*} | *physical-port-name*. The subparameter integer has the format: {*decimal-digit*} | {*integer*}{*decimal-digit*}. (Range for the decimal-digit: 0–9)

The following combinations are possible:

- **ipv6\_address%interface\_id** - Refers to the IPv6 address on the interface specified.
- **ipv6\_address%0** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **ipv6\_address** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **hostname**—Specifies the server hostname. Only translation to IPv4 addresses is supported. (Length: 1–158 characters. Maximum label length: 63 characters)
- **poll**—Enables polling.
- **key keyid**—Specifies the Authentication key to use when sending packets to this peer. (Range: 1–4294967295)

## Default Configuration

No servers are defined.

## Command Mode

Global Configuration mode

## User Guidelines

Up to 8 SNTP servers can be defined.

The `sntp unicast client enable` Global Configuration mode command enables predefined Unicast clients.

The `sntp server` Global Configuration mode command globally enables polling.

The format of an IPv6Z address is: `<ipv6-link-local-address>%<interface-name>`.

**interface-name** = `vlan<integer> / po<integer> / isatap<integer> / <physical-port-name> / 0`. When 0 is specified, the default interface is used.

**integer** = `<decimal-number> / <integer><decimal-number>`

**decimal-number** = `0 / 1 / 2 / 3 / 4 / 5 / 6 / 7 / 8 / 9`

**physical-port-name** = Designated port number, for example: gi6.

If the egress interface is not specified, the default interface is selected.

The following combinations are possible:

- **ipv6\_address%interface\_id** - Refers to the IPv6 address on the interface specified.
- **ipv6\_address%0** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **ipv6\_address** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.

## Example

The following example configures the device to accept SNTP traffic from the server on 192.1.1.1 with polling.

```
Console(config)# sntp server 192.1.1.1 poll
```

## 6.12 show clock

The `show clock` EXEC mode command displays the time and date from the system clock.

**Syntax****show clock** [*detail*]**Parameters****detail**—Displays the TimeZone and SummerTime configuration.**Command Mode**

EXEC mode

**Example**

The following example displays the system time and date.

---

```
Console> show clock
15:29:03 PDT(UTC-7) Jun 17 2002
Time source is SNTP

Console> show clock detail
15:29:03 PDT(UTC-7) Jun 17 2002
Time source is SNTP

Time zone:
Acronym is PST
Offset is UTC-8

Summertime:
Acronym is PDT
Recurring every year.
Begins at first Sunday of April at 2:00.
Ends at last Sunday of October at 2:00.
Offset is 60 minutes.
```

DHCP timezone: Disabled

---

Device> **show clock detail**

15:29:03 PDT(UTC-7) Jun 17 2002

Time source is SNTP

Timezone (DHCP):

Acronym is PST

Offset is UTC-8

Timezone (static):

Acronym is PST

Offset is UTC-8

Summertime (Static):

Acronym is PDT

Recurring every year.

Begins at first Sunday of April at 2:00.

Ends at last Sunday of October at 2:00.

Offset is 60 minutes.

DHCP timezone: Enabled

---

## 6.13 show sntp configuration

The **show sntp configuration** Privileged EXEC mode command displays the Simple Network Time Protocol (SNTP) configuration on the device.

### Syntax

**show sntp configuration**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### Example

The following example displays the device's current SNTP configuration.

---

```
console# show sntp configuration

SNTP port : 123 .

Polling interval: 1024 seconds.

No MD5 authentication keys.

Authentication is not required for synchronization.

No trusted keys.

Unicast Clients: Enabled

Unicast Clients Polling: Enabled

          Server                Polling  Encryption Key
-----
          1.1.1.121             Disabled  Disabled

Broadcast Clients: disabled

Anycast Clients: disabled

No Broadcast Interfaces.

console#
```

---

## 6.14 show sntp status

The **show sntp status** Privileged EXEC mode command displays the Simple Network Time Protocol (SNTP) servers status.

### Syntax

**show sntp status**

**Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays the SNTP servers status.

---

```
Console# show sntp status
```

```
Clock is synchronized, stratum 4, reference is 176.1.1.8, unicast
Reference time is AFE2525E.70597B34 (00:10:22.438 PDT Jul 5 1993)
```

```
Unicast servers:
```

Server	Status	Last response	Offset	Delay
			[mSec]	[mSec]
-----	-----	-----	-----	-----
176.1.1.8	Up	19:58:22.289 PDT Feb 19	7.33	117.79
176.1.8.1	Unknown	2005	8.98	189.19
79		12:17.17.987 PDT Feb 19		
		2005		

```
Anycast server:
```

Server	Interface	Status	Last response	Offset	Delay
				[mSec]	[mSec]
-----	-----	-----	-----	-----	-----
176.1.11.	VLAN 118	Up	9:53:21.789	7.19	119.89
8			PDT Feb 19		
			2005		



Broadcast:

Server	Interface	Last response
-----	-----	-----
176.9.1.1	VLAN 119	19:17:59.792 PDT Feb 19 2002

## 7 Configuration and Image File Commands

### 7.1 copy

The **copy** Privileged EXEC mode command copies a source file to a destination file.

#### Syntax

```
copy source-url destination-url [snmp]
```

#### Parameters

- **source-url**—Specifies the source file URL or source file reserved keyword to be copied. (Length: 1–160 characters)
- **destination-url**—Specifies the destination file URL or destination file reserved keyword. (Length: 1–160 characters).
- **snmp**—Specifies that the destination/source file is in SNMP format. Used only when copying from/to the Startup Configuration file.

The following table displays the URL options.

Keyword	Source or Destination
<b>flash://</b>	Source or destination URL for flash memory. This is the default URL if a URL is specified without a prefix.
<b>running-config</b>	Currently running configuration file. This cannot be the destination file.
<b>startup-config</b> <b>flash://startup-config</b>	Startup configuration file.
<b>image</b> <b>flash://image</b>	Image file. If specified as the source file, it is the active image file. If specified as the destination file, it is the non-active image file.
<b>boot</b>	Boot file.
<b>tftp://</b>	Source or destination URL for a TFTP network server. The syntax for this alias is <i>tftp://host/[directory]/filename</i> . The host can be either an IP address or a host name.

Keyword	Source or Destination
<b>xmodem:</b>	Source for the file from a serial connection that uses the Xmodem protocol.
<b>null:</b>	Null destination for copies or files. A remote file can be copied to null to determine its size. For instance <b>copy running-conf null</b> returns the size of the running configuration file.
<b>backup-config</b>	Backup configuration file. A configuration file can be downloaded to this file (without giving a file name). This can then be copied to the running-conf or startup-conf files.
<b>mirror-config</b>	Mirrored configuration file. If the running config and the startup config have been identical for 24 hours, the startup config is automatically copied to the mirror-conf file by the system. It can then be copied to the startup or running conf if required.
<b>localization</b>	This enables copying a language dictionary file to the secondary language file, such as in <b>copy tftp://10.5.234.203/french.txt localization</b> . This creates French as the second language. the file <b>french.txt</b> is the French dictionary.
<b>logging</b>	Specifies the SYSLOG file.
<b>Word&lt;1-128&gt;</b>	Name of file.

### Command Mode

Privileged EXEC mode

### User Guidelines

The location of a file system dictates the format of the source or destination URL.

The entire copying process may take several minutes and differs from protocol to protocol and from network to network.

If the IPv6 address is a Link Local address (IPv6Z address), the outgoing interface name must be specified. The format of an IPv6Z address is:

*{ipv6-link-local-address}%{interface-id}*. The subparameters are:

- **ipv6-link-local-address**—Specifies the IPv6 Link Local address.
- **interface-id**—{<port-type>[ ]<port-number>}{port-channel | po}[ ]<port-channel-number> | {tunnel | tu}[ ]<tunnel-number> | vlan[ ]<vlan-id>

If the egress interface is not specified, the default interface is selected. The following combinations are possible:

- **ipv6\_address%interface\_id** - Refers to the IPv6 address on the interface specified.
- **ipv6\_address%0** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **ipv6\_address** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.

### Understanding Invalid Combinations of Source and Destination

The following are invalid combinations of source and destination files:

- The source file and destination file are the same file.
- **xmodem:** is the destination file. The source file can be copied to **image**, **boot** and **null:** only.
- **tftp://** is the source file and destination file on the same copy.
- **\*.prv** files cannot be copied.
- The destination file cannot be the Running Configuration file.
- **<for products with mirror-config>mirror-config** cannot be used as a destination

The following table describes the characters displayed by the system when **copy** is being run:

Character	Description
!	For network transfers, indicates that the copy process is taking place. Each exclamation point indicates successful transfer of ten packets (512 bytes each).
.	For network transfers, indicates that the copy process timed out.

### Copying an Image File from a Server to Flash Memory

Use the **copy source-ur/flash://image** command to copy an image file from a server to flash memory. When the administrator copies an image file from the server to a device, the image file is saved to the "inactive" image. To use this image, the administrator must switch the inactive image to the active image and reboot. The device will then use this new image.

### Copying a Boot File from a Server to Flash Memory

Use the **copy *source-url* boot** command to copy a boot file from a server to flash memory. **Copying a Configuration File from a Server to the Startup Configuration**

Use the **copy *source-url* startup-config** command to copy a configuration file from a network server to the device startup configuration file. The startup configuration file is replaced by the copied configuration file.

### Storing the Running Config or Startup Config on a Server

Use the **copy running-config *destination-url*** command to copy the current configuration file to a network server using TFTP.

Use the **copy startup-config *destination-url*** command to copy the startup configuration file to a network server.

### Saving the Running Configuration to the Startup Configuration

Use the **copy running-config startup-config** command to copy the running configuration to the startup configuration file.

### Backing Up the Running Configuration or Startup Configuration to the Backup Configuration

Use the **copy running-config backup-config** command to back up the running configuration to the backup configuration file.

Use the **copy startup-config backup-config** command to back up the startup configuration to the backup configuration file.

### Restoring the Mirror Configuration File.

Use **copy mirror-config startup-config** or **copy mirror-config running-config** to copy the mirror configuration file to one of the configuration files being used.

### Examples

**Example 1** - The following example copies system image file1 from the TFTP server 172.16.101.101 to the non-active image file.

---

```
Console# copy tftp://172.16.101.101/file1 image
```

```
Accessing file 'file1' on 172.16.101.101...
```

```
Loading file1 from 172.16.101.101:
```

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! [OK]

Copy took 0:01:11 [hh:mm:ss]

```

---

### Example 2 - Copying an Image from a Server to Flash Memory

The following example copies a system image named file1 from the TFTP server with an IP address of 172.16.101.101 to a non-active image file.

```

console# copy tftp://172.16.101.101/file1 flash://image

Accessing file 'file1' on 172.16.101.101...

Loading file1 from 172.16.101.101:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! [OK]

Copy took 0:01:11 [hh:mm:ss]

```

---

### Example 3 - Copying the mirror-config file to the startup-configuration file

The following example copies the mirror configuration file, saved by the system, to the Startup Configuration file.

```

console# copy mirror-config startup-config

```

---

## 7.2 write memory

Use the **write memory** Privileged EXEC mode command to save the Running Configuration file to the Startup Configuration file.

### Syntax

**write memory**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### Examples

The following example copies system image file1 from the TFTP server 172.16.101.101 to a non-active image file.

---

```
Console# write memory
```

```
Overwrite file [startup-config] ?[Yes/press any key for no]... 15-Sep-2010 11:27
```

```
:48 %COPY-I-FILECPY: Files Copy - source URL running-config destination URL  
flash://startup-config
```

```
15-Sep-2010 11:27:50 %COPY-N-TRAP: The copy operation was completed  
successfully
```

```
Copy succeeded
```

---

## 7.3 write

Use the **write** Privileged EXEC mode command to save the running configuration to the startup configuration file.

### Syntax

**write**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### Examples

The following example copies system image file1 from the TFTP server 172.16.101.101 to a non-active image file.

---

```
Console# write
Overwrite file [startup-config] ?[Yes/press any key for no]...15-Sep-2010 11:27
:48 %COPY-I-FILECPY: Files Copy - source URL running-config destination URL
flash://startup-config
15-Sep-2010 11:27:50 %COPY-N-TRAP: The copy operation was completed successfully
Copy succeeded
```

---

## 7.4 delete

The **delete** Privileged EXEC mode command deletes a file from a flash memory device.

### Syntax

**delete** *url*



### Parameters

**url**—Specifies the location URL or reserved keyword of the file to be deleted. (Length: 1–160 characters)

The following table displays keywords and URL prefixes:

Keyword	Source or Destination
<b>flash://</b>	URL of the flash memory. This is the default URL if a URL is specified without a prefix.
<b>startup-config</b>	Startup configuration file.
<b>WORD</b>	Name of file.

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### User Guidelines

**\*.sys, \*.prv, image-1 and image-2** files cannot be deleted.

### Example

The following example deletes the file called 'test' from the flash memory.

---

```
Console# delete flash://test
Delete flash:test? [confirm]
```

---

## 7.5 dir

The **dir** Privileged EXEC mode command displays the list of files on a flash file system.

### Syntax

**dir** *[directory-path]*

**Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays the list of files on a flash file system

```
Total size of flash: 33292288 bytes
```

```
Free size of flash: 20708893 bytes
```

```
console# dir
```

```
Directory of flash:
```

File Name	Permission	Flash Size	Data Size	Modified
-----	-----	-----	-----	-----
backuplo	rw	851760	525565	22-Dec-2010 10:50:32
tmp	rw	524288	104	01-Jan-2010 05:35:04
image-1	rw	10485760	10485760	01-Jan-2010 06:10:23
image-2	rw	10485760	10485760	01-Jan-2010 05:43:54
dhcpsn.prv	--	262144	--	01-Jan-2010 05:25:07
sshkeys.prv	--	262144	--	04-Jan-2010 06:05:00
syslog1.sys	r-	524288	--	01-Jan-2010 05:57:00
syslog2.sys	r-	524288	--	01-Jan-2010 05:57:00
directry.prv	--	262144	--	01-Jan-2010 05:25:07
startup-config	rw	786432	1081	01-Jan-2010 10:05:34

```
Total size of flash: 66322432 bytes
```

```
Free size of flash: 42205184 bytes
```

## 7.6 more

The **more** Privileged EXEC mode command displays a file.

### Syntax

**more** *url*

### Parameters

**url**—Specifies the location URL or reserved keyword of the source file to be displayed. (Length: 1–160 characters).

The following table displays options for the URL parameter:

Keyword	Source or Destination
<b>flash://</b>	Source or destination URL for flash memory. If a URL is specified without a prefix, this is the default URL.
<b>running-config</b>	Current running configuration file.
<b>startup-config</b>	Startup configuration file.
<b>mirror-config</b>	Mirrored configuration file.

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### User Guidelines

Files are displayed in ASCII format, except for the images, which are displayed in a hexadecimal format.

**\*.prv** files cannot be displayed.

### Example

The following example displays the running configuration file contents.

```
console# more running-config  
no spanning-tree
```

```
interface range gil-48
speed 1000
exit
no lldp run
line console
exec-timeout 0
```

---

## 7.7 boot system

The **boot system** Privileged EXEC mode command specifies the active system image file that is loaded by the device at startup.

### Syntax

**boot system** *{image-1 | image-2}*

### Parameters

- **image-1**—Specifies that image-1 is loaded as the system image during the next device startup.
- **image-2**—Specifies that image-2 is loaded as the system image during the next device startup.

### Default Configuration

This command has no default configuration.

### Command Mode

Privileged EXEC mode

### User Guidelines

Use the [show bootvar](#) command to display the active image.

### Example

The following example specifies that **image-1** is the active system image file loaded by the device at startup.

---

```
Console# boot system image-1
```

---

## 7.8 show bootvar

Use the **show bootvar** EXEC mode command to display the active system image file that is loaded by the device at startup.

### Syntax

**show bootvar**

### Parameters

N/A

### Command Mode

EXEC mode

### Example

The following example displays the active system image file that is loaded by the device at startup.

---

```
Console# show bootvar

Image      filename      Version      Date              Status
-----      -
1          image-1       1.1.04       23-Jul-2010      -
2          image-2       1.1.0.5      22-Jan-2010      Active
                                           Not active*
```

\*\*\*: Designates that the image was selected for the next boot.

---

## 7.9 show running-config

The **show running-config** Privileged EXEC mode command displays the entire current Running Configuration file contents or the contents of the file for the specified interface(s).

### Syntax

**show running-config** [*interface interface-id-list*]

### Parameters

***interface interface-id-list***—Specifies a list of interface IDs. The interface IDs can be one of the following types: Ethernet port, Port-channel or VLAN.

### Command Mode

Privileged EXEC mode

### User Guidelines

The Running Configuration file does not contain all the information that can be displayed in the output. Only non-default configurations are displayed.

### Example

The following example displays the Running Configuration file contents.

**Example 1** - Show the entire Running Configuration file.

---

```
Console# show running-config
no spanning-tree
interface range gil-48
speed 1000
exit
no lldp run
interface vlan 1
ip address 1.1.1.1 255.0.0.0
exit
line console
exec-timeout 0
exit
console#
```

---

**Example 2** - Show the entire Running Configuration file for ports 1 and 2.

```
console# show running-config interface fastethernet 1-2
interface fastethernet 1
back-pressure
```

```
duplex half
speed 10
flowcontrol on
negotiation 10h 100h 100f
dot1x max-req 8
description "Hello World String"
lacp timeout short
lacp port-priority 1234
garp timer join 100
garp timer leave 300
port security max 111
port security mode max-addresses
spanning-tree disable
spanning-tree portfast auto
spanning-tree link-type point-to-point
spanning-tree cost 200000
spanning-tree port-priority 224
spanning-tree guard root
spanning-tree mst 2 port-priority 64
spanning-tree mst 2 cost 2222
spanning-tree mst 4 port-priority 80
qos cos 6
traffic-shape 12345
switchport mode general
switchport general allowed vlan add 12,14-20 tagged
switchport general allowed vlan add 2-11,13,100,3000,3002,3004,3006,3008
untagged
switchport general map macs-group 1 vlan 111
switchport general ingress-filtering disable
switchport general acceptable-frame-type untagged-only
switchport general pvid 111
```

```
interface fastethernet 2

ip address 1.100.100.100 255.0.0.0
switchport mode trunk
switchport general map macs-group 1 vlan 111
switchport general map subnets-group 1 vlan 113
switchport general map protocols-group 1 vlan 112
switchport general ingress-filtering disable
switchport general acceptable-frame-type untagged-only
switchport general pvid 111
switchport trunk native vlan 22
```

---

## 7.10 show startup-config

The **show startup-config** Privileged EXEC mode command displays the startup configuration file contents.

### Syntax

```
show startup-config [interface interface-id-list]
```

### Parameters

**interface** *interface-id-list*—Specifies list of interface IDs. The interface IDs can be one of the following types: Ethernet port, Port-channel or VLAN.

### Command Mode

Privileged EXEC mode

### User Guidelines

The Startup Configuration file does not contain all the information that can be displayed in the output. Only non-default configurations are displayed.



## Example

**Example 1** - The following example displays the Startup Configuration file contents.

---

```
Console# show startup-config

no spanning-tree

interface range gil-48

speed 1000

exit

no lldp run

interface vlan 1

ip address 1.1.1.1 255.0.0.0

exit

line console

exec-timeout 0

exit

console#
```

---

**Example 2** - The following example displays the Startup Configuration file contents for ports 1 and 2.

```
console# show startup-config interface gil-2

interface gil

back-pressure

duplex half

speed 10

flowcontrol on

negotiation 10h 100h 100f

dot1x max-req 8

description "Hello World String"

lACP timeout short

lACP port-priority 1234
```

```
garp timer join 100
garp timer leave 300
port security max 111
port security mode max-addresses
spanning-tree disable
spanning-tree portfast auto
spanning-tree link-type point-to-point
spanning-tree cost 200000
spanning-tree port-priority 224
spanning-tree guard root
spanning-tree mst 2 port-priority 64
spanning-tree mst 2 cost 2222
spanning-tree mst 4 port-priority 80
qos cos 6
traffic-shape 12345
switchport mode general
switchport general allowed vlan add 12,14-20 tagged
switchport general allowed vlan add 2-11,13,100,3000,3002,3004,3006,3008
untagged
switchport general map macs-group 1 vlan 111
switchport general ingress-filtering disable
switchport general acceptable-frame-type untagged-only
switchport general pvid 111
interface fastethernet 2
ip address 1.100.100.100 255.0.0.0
switchport mode trunk
switchport general map macs-group 1 vlan 111
switchport general map subnets-group 1 vlan 113
switchport general map protocols-group 1 vlan 112
switchport general ingress-filtering disable
switchport general acceptable-frame-type untagged-only
```

---

```
switchport general pvid 111
switchport trunk native vlan 22
```

---

## 8 Auto-Configuration

---

### 8.1 boot host auto-config

Use the **boot host auto-config** Global Configuration mode command to enable auto configuration via DHCP. Use the **no** form of this command to disable DHCP auto configuration.

#### Syntax

**boot host auto-config**

**no boot host auto-config**

#### Parameters

N/A

#### Default Configuration

Enabled by default.

#### Command Mode

Global Configuration mode

#### Default Configuration

Enabled by default.

#### Example

---

```
console(conf)# boot host auto-config
```

---

### 8.2 show boot

Use the **show boot** Privilege EXEC mode command to show the status of the IP DHCP Auto Config process.

#### Syntax

**show boot**

**Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

Privilege EXEC mode

**Examples**

---

```
console# show boot
Auto Config
-----
Config Download via DHCP: enabled
Next Boot Config Download via DHCP: default
```

---

### 8.3 ip dhcp tftp-server ip address

Use the **ip dhcp tftp-server ip address** Global Configuration mode command to set the TFTP server's IP address. This address server as the default address used by a switch when it has not been received from the DHCP server.

Use the **no** form of this command to remove the address.

**Syntax****ip dhcp tftp-server ip address** *ip-addr***no ip dhcp tftp-server ip address****Parameters****ip addr** *ip-addr*—Address of TFTP server**Default Configuration**

No IP address

### Command Mode

Global Configuration mode

### Examples

---

```
console(conf)# ip dhcp tftp-server ip address 10.5.234.232
```

---

## 8.4 ip dhcp tftp-server file

Use the **ip dhcp tftp-server file** Global Configuration mode command to set the full file name of the configuration file to be downloaded on the TFTP server when it has not been received from the DHCP server. This serves as the default configuration file.

Use the **no** form of this command to remove the name.

### Syntax

```
ip dhcp tftp-server file file-path
```

```
no ip dhcp tftp-server file
```

### Parameters

**file-path**—Full file path and name of the configuration file on TFTP server

### Default Configuration

No file name

### Command Mode

Global Configuration mode

### Examples

---

```
console(conf)# ip dhcp tftp-server file conf/conf-file
```

---

## 8.5 show ip dhcp tftp-server

Use the **show ip dhcp tftp-server EXEC** mode command to display information about the TFTP server.

**Syntax****show ip dhcp tftp-server****Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

EXEC

**Example**

---

```
console# show ip dhcp tftp server

tftp server address

active      1.1.1.1 from sname

manual     2.2.2.2

file path on tftp server

active     conf/conf-file from option 67
```

---

## 8.6 ip dhcp information option

Use the **ip dhcp information option** Global Configuration command to enable DHCP option-82 data insertion. Use the **no** form of this command to disable DHCP option-82 data insertion.

**Syntax****ip dhcp information option****no ip dhcp information option****Parameters**

N/A

**Default Configuration**

DHCP option-82 data insertion is disabled.

**Command Mode**

Global Configuration mode

**User Guidelines**

DHCP option 82 would be enabled only if DHCP snooping or DHCP relay are enabled.

**Example**

---

```
console(config)# ip dhcp information option
```

---

## 8.7 show ip dhcp information option

The **show ip dhcp information option** EXEC mode command displays the DHCP Option 82 configuration.

**Syntax**

**show ip dhcp information option**

**Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

EXEC mode

**Example**

The following example displays the DHCP Option 82 configuration.

---

```
console# show ip dhcp information option
Relay agent Information option is Enabled
```

---



## 9 Management ACL Commands

### 9.1 management access-list

The **management access-list** Global Configuration mode command configures a management access list (ACL) and enters the Management Access-List Configuration command mode. Use the **no** form of this command to delete an ACL

#### Syntax

**management access-list** *name*

**no management access-list** *name*

#### Parameters

**name**—Specifies the ACL name. (Length: 1–32 characters)

#### Default Configuration

N/A

#### Command Mode

Global Configuration mode

#### User Guidelines

Use this command to configure a management access list. This command enters the Management Access-List Configuration mode, where the denied or permitted access conditions are defined with the **deny** and **permit** commands.

If no match criteria are defined, the default value is **deny**.

When re-entering the access-list context, the new rules are entered at the end of the access list.

Use the [management access-class](#) command to select the active access list.

The active management list cannot be updated or removed.

For IPv6 management traffic that is tunneled in IPv4 packets, the management ACL is applied first on the external IPv4 header (rules with the service field are ignored), and then again on the inner IPv6 header.

## Example

**Example 1** - The following example creates a management access list called **m1ist**, configures management **gi 1** and **gi9**, and makes the new access list the active list.

---

```
Console(config)# management access-list m1ist
Console(config-macl)# permit gi1
Console(config-macl)# permit gi9
Console(config-macl)# exit
Console(config)# management access-class m1ist
```

---

**Example 2** - The following example creates a management access list called 'm1ist', configures all interfaces to be management interfaces except **gi1** and **9**, and makes the new access list the active list.

---

```
Console(config)# management access-list m1ist
Console(config-macl)# deny gi1
Console(config-macl)# deny gi9
Console(config-macl)# permit
Console(config-macl)# exit
Console(config)# management access-class m1ist
```

---

## 9.2 permit (Management)

The **permit** Management Access-List Configuration mode command sets permit rules (ACEs) for the management access list (ACL).

### Syntax

**permit** *[interface-id] [service service]*

**permit ip-source** {*ipv4-address* | *ipv6-address* | *ipv6-prefix-length*} [*mask {mask / prefix-length}*] *[interface-id] [service service]*

### Parameters

- **interface-id:**—Specify an interface ID. The interface ID can be one of the following types: Ethernet port, Port-channel or VLAN
- **service *service*** — Specifies the service type. Possible values are: Telnet, SSH, HTTP, HTTPS and SNMP.
- **ipv4-address**— Specifies the source IPv4 address.
- **ipv6-address/ipv6-prefix-length**— Specifies the source IPv6 address and source IPv6 address prefix length. The prefix length must be preceded by a forward slash (/). The parameter is optional.
- **mask *mask*** — Specifies the source IPv4 address network mask. This parameter is relevant only to IPv4 addresses.
- **mask *prefix-length*** — Specifies the number of bits that comprise the source IPv4 address prefix. The prefix length must be preceded by a forward slash (/). This parameter is relevant only to IPv4 addresses. (Range: 0–32)

### Default Configuration

No rules are configured.

### Command Mode

Management Access-List Configuration mode

### User Guidelines

Rules with Ethernet, VLAN, and port-channel parameters are valid only if an IP address is defined on the appropriate interface.

### Example

The following example permits all ports in the ACL called **m1ist**

---

```
Console(config)# management access-list m1ist
Console(config-macl)# permit
```

---

## 9.3 deny (Management)

The **deny** Management Access-List Configuration mode command sets permit rules (ACEs) for the management access list (ACL).

## Syntax

**deny** [*interface-id*] [**service** *service*]

**deny ip-source** {*ipv4-address* | *ipv6-address/ipv6-prefix-length*} [**mask** {*mask* | *prefix-length*}] [*interface-id*] [**service** *service*]

## Parameters

- **interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port, Port-channel or VLAN
- **service** *service*—Specifies the service type. Possible values are: Telnet, SSH, HTTP, HTTPS and SNMP.
- **ipv4-address**—Specifies the source IPv4 address.
- **ipv6-address/ipv6-prefix-length**—Specifies the source IPv6 address and source IPv6 address prefix length. The prefix length must be preceded by a forward slash (/). The parameter is optional.
- **mask** *mask*—Specifies the source IPv4 address network mask. The parameter is relevant only to IPv4 addresses.
- **mask** *prefix-length*—Specifies the number of bits that comprise the source IPv4 address prefix. The prefix length must be preceded by a forward slash (/). The parameter is relevant only to IPv4 addresses. (Range: 0–32)

## Default Configuration

No rules are configured.

## Command Mode

Management Access-List Configuration mode

## User Guidelines

Rules with ethernet, VLAN, and port-channel parameters are valid only if an IP address is defined on the appropriate interface.

## Example

The following example denies all ports in the ACL called **m1ist**.

---

```
Console(config)# management access-list m1ist
Console(config-macl)# deny
```

---

## 9.4 management access-class

The **management access-class** Global Configuration mode command restricts management connections by defining the active management access list (ACL). To disable management connection restrictions, use the **no** form of this command.

### Syntax

**management access-class** {**console-only** | *name*}

**no management access-class**

### Parameters

- **console-only**—Specifies that the device can be managed only from the console.
- **name**—Specifies the ACL name to be used. (Length: 1–32 characters)

### Default Configuration

The default configuration is no management connection restrictions.

### Command Mode

Global Configuration mode

### Example

The following example defines an access list called **m1ist** as the active management access list.

---

```
Console(config)# management access-class m1ist
```

---

## 9.5 show management access-list

The **show management access-list** Privileged EXEC mode command displays management access lists (ACLs).

### Syntax

**show management access-list** [*name*]

### Parameters

**name**—Specifies the name of a management access list to be displayed. (Length: 1–32 characters)

### Default Configuration

All management ACLs are displayed.

### Command Mode

Privileged EXEC mode

### Example

The following example displays the **mlist** management ACL.

---

```
Console# show management access-list mlist
console-only
-----
deny
! (Note: all other access implicitly denied)
mlist
-----
permit gil
permit gi9
! (Note: all other access implicitly denied)
console#
```

---

## 9.6 show management access-class

The **show management access-class** Privileged EXEC mode command displays information about the active management access list (ACLs).

### Syntax

**show management access-class**

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays the active management ACL information.

---

```
Console# show management access-class
```

```
Management access-class is enabled, using access list mlist
```

---

# 10 Network Management Protocol (SNMP) Commands

---

## 10.1 snmp-server

Use the **snmp-server server** Global Configuration mode command to enable the device to be configured by SNMP. Use the **no** form of this command to disable this function.

### Syntax

**snmp-server server**

**no snmp-server server**

### Parameters

N/A

### Default Configuration

Enabled

### Command Mode

Global Configuration mode

### Example

---

```
console(config)# snmp-server server
```

---

## 10.2 snmp-server community

Use the **snmp-server community** Global Configuration mode command to set up the community access string to permit access to the Simple Network Management Protocol command. Use the **no** form of this command to remove the specified community string.

### Syntax

**snmp-server community** *string* [**view** *view-name*] [**ro** | **rw** | **su**] [*ipv4-address* / *ipv6-address*] [**mask** | **prefix-length**] [**type** *router*]

**snmp-server community-group** *string* *group-name* [*ipv4-address* / *ipv6-address*] [**mask** | **prefix-length**] [**type** *router*]



**no snmp-server community** *string [ipv4-address/ipv6-address]*

### Parameters

- **string**—Community string that acts like a password and permits access to the SNMP protocol. (Range: 1–20 characters)
- **ro**—Specifies read-only access (default)
- **rw**—Specifies read-write access
- **su**—Specifies SNMP administrator access
- **view** *view-name*—Specifies the name of a view to be configured using the command **snmp-server view** (no specific order of the command configurations is imposed on the user). The view defines the objects available to the community. It is not relevant for **su**, which has access to the whole MIB. If unspecified, all the objects, except the community-table and SNMPv3 user and access tables, are available. (Range: 1–30 characters)
- **ipv4-address**—Management station IPv4 address. The default is all IP addresses.
- **ipv6-address**—Management station IPv4 address. The default is all IP addresses.

The following combinations are possible:

- **ipv6\_address%interface\_id** - Refers to the IPv6 address on the interface specified.
- **ipv6\_address%0** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **ipv6\_address** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **mask**—Specifies the mask of the IPv4 address. This is not a network mask, but rather a mask that defines which bits of the packet's source address are compared to the configured IP address. If unspecified, it defaults to 255.255.255.255. The command returns an error if the mask is specified without an IPv4 address.
- **prefix-length**—Specifies the number of bits that comprise the IPv4 address prefix. If unspecified, it defaults to 32. The command returns an error if the prefix-length is specified without an IPv4 address.

- **group-name**—Specifies the name of a group that should be configured using the command **snmp-server group** with v1 or v2 parameter (no specific order of the two command configurations is imposed on the user). The group defines the objects available to the community. (Range: 1–30 characters)
- **type router**—Specifies that SNMP requests for duplicate tables configure the router tables. This is the default.

### Default Configuration

No community is defined

### Command Mode

Global Configuration mode

### User Guidelines

You cannot specify a view-name for su, which has access to the whole MIB.

You can use the view-name to restrict the access rights of a community string.

The logical key of the command is the pair (community, ip-address). If ip-address is omitted then the key is (community, All-Ips).

By specifying the view-name parameter, the software:

- Generates an internal security-name.
- Maps the internal security-name for SNMPv1 and SNMPv2 security models to an internal group-name.
- Maps the internal group-name for SNMPv1 and SNMPv2 security models to view-name (read-view and notify-view always, and for rw for write-view also),

You can use the group-name to restrict the access rights of a community string. By specifying the group-name parameter the software:

- Generates an internal security-name.
- Maps the internal security-name for SNMPv1 and SNMPv2 security models to the group-name.

The **snmp-server community-group** command and **snmp-server user** command for v1 and v2 are equivalent. You should use the **snmp-server community-group** command when you want to configure the ipv4-address|ipv6-address management addresses.

The `Type` keyword is used for a different purpose. Therefore, when defining an SNMP community, the administrator must indicate which tables are being configured. If `Type` is `router`, it means that the device's tables are being configured.

### Example

```
console(config)# snmp-server community abcd su 1.1.1.121 mask 255.0.0.0
console(config)# snmp-server community-group tom abcd 1.1.1.122 prefix 8
```

## 10.3 snmp-server view

The `snmp-server view` Global Configuration mode command creates or updates a Simple Network Management Protocol (SNMP) server view entry. Use the `no` form of this command to remove an SNMP server view entry.

### Syntax

```
snmp-server view view-name oid-tree [included / excluded]
```

```
no snmp-server view view-name [oid-tree]
```

### Parameters

- **view-name**—Specifies the label for the view record that is being created or updated. The name is used to reference the record. (Length: 1–30 characters)
- **oid-tree**—Specifies the ASN.1 subtree object identifier to be included or excluded from the view. To identify the subtree, specify a text string consisting of numbers, such as 1.3.6.2.4, or a word, such as System. Replace a single sub-identifier with the asterisk (\*) wildcard to specify a subtree family; for example 1.3.\*.4.
- **included**—Specifies that the view type is included.
- **excluded**—Specifies that the view type is excluded.

### Default Configuration

The following views are created by default:

- **Default** - Contains all MIBs except for those that configure the SNMP parameters themselves.
- **DefaultSuper** - Contains all MIBs.

## Command Mode

Global Configuration mode

## User Guidelines

This command can be entered multiple times for the same view record.

The command logical key is the pair (view-name, oid-tree).

The number of views is limited to 64.

Default and DefaultSuper views are reserved for internal software use and cannot be deleted or modified.

## Example

The following example creates a view that includes all objects in the MIB-II system group except for sysServices (System 7) and all objects for interface 1 in the MIB-II interface group.

```
Console(config)# snmp-server view user-view system included
Console(config)# snmp-server view user-view system.7 excluded
Console(config)# snmp-server view user-view ifEntry.*.1 included
```

## 10.4 snmp-server group

The **snmp-server group** Global Configuration mode command configures a new Simple Network Management Protocol (SNMP) group or a table that maps SNMP users to SNMP views. Use the **no** form of this command, remove a specified SNMP group.

### Syntax

```
snmp-server group groupname {v1 | v2 | v3 [noauth | auth | priv]} [notify notifyview]
[read readview] [write writeview]
```

```
no snmp-server group groupname {v1 | v2 | v3 [noauth | auth | priv]} [context name]
```

### Parameters

- **groupname**—Specifies the group name. (Length: 1–30 characters)
- **v1**—Specifies the SNMP Version 1 security model.
- **v2**—Specifies the SNMP Version 2 security model.

- **v3**—Specifies the SNMP Version 3 security model.
- **noauth**—Specifies no packet authentication. Applicable only to the SNMP version 3 security model.
- **auth**—Specifies packet authentication without encryption. Applicable only to the SNMP version 3 security model.
- **priv**—Specifies packet authentication with encryption. Applicable only to the SNMP version 3 security model.
- **notify notifyview**—Specifies the view name that enables specifying an inform or a trap. Applicable only to the SNMP version 3 security model. (Length: 1–30 characters)
- **read readview**—Specifies the view name that enables viewing only the agent contents. (Length: 1–30 characters)
- **write writeview**—Specifies the view name that enables entering data and configuring the agent contents. (Length: 1–30 characters)

### Default Configuration

No group entry exists.

If **notifyview** is not specified, nothing is defined for the notify view.

If **readview** is not specified, all objects except for the community-table and SNMPv3 user and access tables are available.

If **writeview** is not specified, nothing is defined for the write view.

### Command Mode

Global Configuration mode

### User Guidelines

The command logical key is (**groupname, snmp-version, security-level**). For snmp-version v1/v2 the security-level is always **noauth**.

### Example

The following example attaches a group called user-group to SNMPv3 and assigns to the group the privacy security level and read access rights to a view called user-view.

```
Console(config)# snmp-server group user-group v3 priv read user-view
```

---

## 10.5 snmp-server user

Use the **snmp-server user** Global Configuration mode command to configure a new SNMP Version 3 user. Use the **no** form of the command to remove a user.

### Syntax

```
snmp-server user username groupname {v1 | v2c | [remote-host] v3 [auth {md5 / sha} auth-password]}
```

```
no snmp-server user username [remote host]
```

### Parameters

- **username**—The name of the user on the host that connects to the agent. (Range: Up to 20 characters)
- **groupname**—The name of the group to which the user belongs. The group should be configured using the command **snmp-server group** with v3 parameters (no specific order of the 2 command configurations is imposed on the user). (Range: Up to 30 characters)
- **remote host**—IP address of the remote SNMP host.
- **v1**—Specifies that v1 is to be used.
- **v2c**—Specifies that v2c is to be used.
- **v3**—Specifies that v3 is to be used.
- **auth**—Specifies which authentication level is to be used.
- **md5**—Specifies the HMAC-MD5-96 authentication level.
- **Sha**—Specifies the HMAC-SHA-96 authentication level.
- **auth-password**—Specifies the authentication and privacy password (the same password is used for both) password.

**Parameters Range auth password** - Up to 32 characters.

### Default Configuration

No group entry exists.

## Command Mode

Global configuration

## User Guidelines

If **auth md5** or **auth sha** is specified, both authentication and privacy are enabled for the user.

When you enter a **show running-config** command, you do not see a line for this user. To see if this user has been added to the configuration, type the **show snmp user** command.

An SNMP EngineID should be defined in order to add users to the device (in the **snmp-server engineid local** or **remote** commands).

Changing or removing the value of **snmpEngineID** deletes the SNMPv3 users' database.

The logical key of the command is Username.

Configuring a remote host is required in order to send informs to that host. A configured remote host is also able to manage the device (besides getting the informs)

To configure a remote user, specify the IP address for the remote SNMP agent of the device where the user resides. Also, before you configure remote users for a particular agent, configure the SNMP engine ID, using the **snmp-server engineid remote** command. The remote agent's SNMP engine ID is needed when computing the authentication and privacy digests from the password. If the remote engine ID is not configured first, the configuration command fails.

Since the same group may be defined several times, each time with different version or different access level (noauth, auth or auth & priv), when defining a user it is not sufficient to specify the group name, rather you must specify group name, version and access level for complete determination of how to handle packets from this user.

## Example

```
console(config)# snmp-server user tom acbd v1
console(config)# snmp-server user tom acbd v2c
console(config)# snmp-server engineid local default
The engine-id must be unique within your administrative domain.
Do you wish to continue? [Y/N]
```

Y

The SNMPv3 database will be erased. Do you wish to continue? [Y/N]

Y

```
console(config)# snmp-server user tom acbd v3
```

---

## 10.6 snmp-server filter

The **snmp-server filter** Global Configuration mode command creates or updates a Simple Network Management Protocol (SNMP) server filter entry. Use the **no** form of this command to remove the specified SNMP server filter entry.

### Syntax

```
snmp-server filter filter-name oid-tree [included | excluded]
```

```
no snmp-server filter filter-name [oid-tree]
```

### Parameters

- **filter-name**—Specifies the label for the filter record that is being updated or created. The name is used to reference the record. (Length: 1–30 characters)
- **oid-tree**—Specifies the ASN.1 subtree object identifier to be included or excluded from the view. To identify the subtree, specify a text string consisting of numbers, such as 1.3.6.2.4, or a word, such as System. Replace a single sub-identifier with the asterisk (\*) wildcard to specify a subtree family; for example, 1.3\*.4.
- **included**—Specifies that the filter type is included.
- **excluded**—Specifies that the filter type is excluded.

### Default Configuration

No view entry exists.

### Command Mode

Global Configuration mode



### User Guidelines

This command can be entered multiple times for the same filter record. If an object identifier is included in two or more lines, later lines take precedence. The command's logical key is the pair (filter-name, oid-tree).

### Example

The following example creates a filter that includes all objects in the MIB-II system group except for sysServices (System 7) and all objects for interface 1 in the MIB-II interfaces group.

```
Console(config)# snmp-server filter f1 system included
Console(config)# snmp-server filter f2 system.7 excluded
Console(config)# snmp-server filter f3 ifEntry.*.1 included
```

## 10.7 snmp-server host

Use the **snmp-server host** Global Configuration mode command to specify the recipient of a Simple Network Management Protocol notification operation. Use the **no** form of this command to remove the specified host.

### Syntax

```
snmp-server host {ipv4-address / ipv6-address / hostname} [traps / informs] [version
{1 | 2c | 3} [auth / noauth / priv]] community-string [udp-port port] [filter filtername]
[timeout seconds] [retries retries]
```

```
no snmp-server host {ipv4-address / ipv6-address / hostname} [traps / informs]
[version {1 | 2c | 3}]
```

### Parameters

- **pv4-address**—IPv4 address of the host (the targeted recipient).
- **ipv6-address**—Pv6 address of the host (the targeted recipient). When the IPv6 address is a Link Local address (IPv6Z address), the outgoing interface name must be specified. Refer to the User Guidelines for the interface name syntax.
- **hostname**—Hostname of the host. (Range: 1–158 characters. Maximum label size: 63)
- **trap**—Sends SNMP traps to this host (default).

- **informs**—Sends SNMP informs to this host. Not applicable to SNMPv1.
- **1**—SNMPv1 traps are used.
- **2c**—SNMPv2 traps are used
- **3**—SNMPv2 traps are used
- **community-string**—Password-like community string sent with the notification operation. (Range: 1–20 characters)
- **noauth**—Specifies no authentication of a packet.
- **auth**—Specifies authentication of a packet without encrypting it.
- **priv**—Specifies authentication of a packet with encryption.
- **udp-port** *port*—UDP port of the host to use. The default is 162. (Range: 1–65535)
- **filter** *filtername*—A string that is the name of the filter that defines the filter for this host. If unspecified, nothing is filtered. The filter should be defined using the command **snmp-server filter** (no specific order of the command configurations is imposed on the user). (Range: Up to 30 characters)
- **timeout** *seconds*—Number of seconds to wait for an acknowledgment before resending informs. The default is 15 seconds. The parameter is relevant only for informs. (Range: 1–300)
- **retries** *retries*—Maximum number of times to resend an inform request, when a response is not received for a generated message. The default is 3. The parameter is relevant only for informs. (Range: 0–255)

### Command Mode

Global Configuration mode

### User Guidelines

The logical key of the command is the pair (ip-address/hostname, traps/informs, version).

When configuring **snmp v1** or **v2** notifications recipient, the software automatically generates a notification view for that recipient for all MIBs. For **SNMPv3** the software doesn't automatically create a user nor a notify view. Use the commands **snmp-server user**, **snmp-server group** and **snmp-server view** in Global Configuration mode to create a user, a group or a notify group respectively.

The format of an IPv6Z address is: `<ipv6-link-local-address>%<interface-name>`

**interface-name** = *vlan*<integer> / *ch*<integer> / *isatap*<integer> /  
 <physical-port-name> / 0

**integer** = <decimal-number> / <integer><decimal-number>

**decimal-number** = 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

**physical-port-name** = Designated port number, for example *gi1/1/16*

The following combinations are possible:

- **ipv6\_address%interface\_id** - Refers to the IPv6 address on the interface specified.
- **ipv6\_address%0** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **ipv6\_address** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.

If the egress interface is not specified, the default interface is selected. Specifying *interface zone=0* is equal to not defining an egress interface.

### Example

The following defines a host at the IP address displayed.

---

```
console(config)# snmp-server host 1.1.1.121 abc
```

---

## 10.8 snmp-server engineID remote

To specify the Simple Network Management Protocol (SNMP) engine ID of a remote SNMP device, use the **snmp-server engineID remote** Global Configuration mode command. Use the **no** form of this command to remove the configured engine ID.

### Syntax

**snmp-server engineID remote** {*ipv4-ip-address* | *ipv6 address*} *engineid-string*

**no snmp-server engineID remote** {*ipv4-ip-address* | *ipv6 address*}

### Parameters

- **ipv4-ip-address** | **ipv6 address**—Pv4 or IPv6 address of the remote device

- **engineid-string**—The character string that identifies the engine ID. The engine ID is a concatenated hexadecimal string. Each byte in hexadecimal character strings is two hexadecimal digits. Each byte can be separated by a period or colon. If the user enters an odd number of hexadecimal digits, the system automatically prefixes the hexadecimal string with a zero. (Range: engineid-string5–32 characters. 9–64 hexadecimal digits)

### Default Configuration

The EngineID is not configured.

### Command Mode

Global Configuration mode

### User Guidelines

A remote engine ID is required when an SNMP version 3 inform is configured. The remote engine ID is used to compute the security digest for authenticating and encrypting packets sent to a user on the remote host.

The format of an IPv6Z address is: *<ipv6-link-local-address>%<interface-name>*

**interface-name** = *vlan<integer> | ch<integer> | isatap<integer> | <physical-port-name> | 0*

**integer** = *<decimal-number> | <integer><decimal-number>*

**decimal-number** = 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

**physical-port-name** = Designated port number, for example *gi1/1/16*

The following combinations are possible:

- **ipv6\_address%interface\_id** - Refers to the IPv6 address on the interface specified.
- **ipv6\_address%0** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **ipv6\_address** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.

## 10.9 snmp-server enable traps

Use the **snmp-server enable traps** Global Configuration mode command to enable the device to send SNMP traps. Use the **no** form of the command to disable SNMP traps.

**Syntax****snmp-server enable traps****no snmp-server enable traps****Default Configuration**

SNMP traps are enabled.

**Command Mode**

Global Configuration mode

**Example**

The following example enables SNMP traps.

---

```
Console(config)# snmp-server enable traps
```

---

**10.10 snmp-server trap authentication**

Use the **snmp-server trap authentication** Global Configuration mode command to enable the device to send SNMP traps when authentication fails. Use the **no** form of this command to disable SNMP failed authentication traps.

**Syntax****snmp-server trap authentication****no snmp-server trap authentication****Default Configuration**

SNMP failed authentication traps are enabled.

**Command Mode**

Global Configuration mode

**Example**

The following example enables SNMP failed authentication traps.

---

```
Console(config)# snmp-server trap authentication
```

---

---

## 10.11 snmp-server contact

Use the **snmp-server contact** Global Configuration mode command to configure the system contact (sysContact) string. Use the **no** form of the command to remove the system contact information.

### Syntax

**snmp-server contact** *text*

**no snmp-server contact**

### Parameters

**text**—Specifies the string describing system contact information. (Length: 1–160 characters)

### Command Mode

Global Configuration mode

### Example

The following example configures the system contact point called Technical\_Support.

---

```
Console(config)# snmp-server contact Technical_Support
```

---

## 10.12 snmp-server location

Use the **snmp-server location** Global Configuration mode command to configure the system location string. Use the **no** form of this command to remove the location string.

### Syntax

**snmp-server location** *text*

**no snmp-server location**

### Parameters

**text**—Specifies a string describing system location information. (Length: 1–160 characters)

### Command Mode

Global Configuration mode

### Example

The following example defines the device location as New\_York.

---

```
Console(config)# snmp-server location New_York
```

---

## 10.13 snmp-server set

Use the **snmp-server set** Global Configuration mode command to define the SNMP MIB value.

### Syntax

```
snmp-server set variable-name name value [name2 value2 ...]
```

### Parameters

- **variable-name**—Specifies the SNMP MIB variable name, which must be a valid string.
- **name value**—Specifies a list of name and value pairs. Each name and value must be a valid string. In the case of scalar MIBs, there is only a single name-value pair. In the case of an entry in a table, there is at least one name-value pair, followed by one or more fields.

### Command Mode

Global Configuration mode

### User Guidelines

Although the CLI can set any required configuration, there might be a situation where an SNMP user sets a MIB variable that does not have an equivalent command. To generate configuration files that support those situations, use the **snmp-server set** command.

**Example**

The following example configures the scalar MIB sysName with the value TechSupp.

---

```
Console(config)# snmp-server set sysName sysname TechSupp
```

---

**10.14 show snmp**

Use the **show snmp** Privileged EXEC mode command to display the SNMP status.

**Syntax**

**show snmp**

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays the SNMP communications status.

```
Console# show snmp
```

```
SNMP is enabled
```

Community-String	Community-Access	View name	IP Address	Type
public	read only	user-view	All	Router
private	read write	Default	172.16.1.1/10	Router
private	su	DefaultSuper	172.16.1.1	Router
Community-string	Group name	IP Address	Type	
public	user-group	All	Router	

```
Traps are enabled.
```

```
Authentication trap is enabled.
```

```
Version 1,2 notifications
```



```

Target Address      Type      Community      Version      UDP      Filter      TO      Retries
                   Port      Name          Sec          Name
-----
192.122.173.42     Trap      public         2            162     15         3
192.122.173.42     Inform   public         2            162     15         3

Version 3 notifications
Target Address      Type      Username      Security      UDP      Filter      TO      Retries
                   Port      name          Level         Port      name          Sec
-----
192.122.173.42     Inform   Bob           Priv          162     15         3

System Contact: Robert

System Location: Marketing

```

The following table describes the significant fields shown in the display.

Field	Description
<b>Community-string</b>	The community access string permitting access to the SNMP protocol.
<b>Community-access</b>	The access type—read-only, read-write, super access.
<b>IP Address</b>	The management station IP Address.
<b>Trap-Rec-Address</b>	The targeted recipient.
<b>Trap-Rec-Community</b>	The statistics sent with the notification operation.
<b>Version</b>	The SNMP version (1 or 2) for the sent trap.

## 10.15 show snmp engineID

Use the **show snmp engineID** Privileged EXEC mode command to display the local Simple Network Management Protocol (SNMP) engine ID.

### Syntax

```
show snmp engineID
```

### Command Mode

Privileged EXEC mode

**Example**

The following example displays the SNMP engine ID.

---

```

Console # show snmp engineID

Local SNMP engineID: 08009009020C0B099C075878

IP address          Remote SNMP engineID
-----
172.16.1.1         08009009020C0B099C075879

```

---

**10.16 show snmp views**

Use the **show snmp views** Privileged EXEC mode command to display the configured SNMP views.

**Syntax**

```
show snmp views [viewname]
```

**Parameters**

**viewname**—Specifies the view name. (Length: 1–30 characters)

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays the configured SNMP views.

---

```

Console# show snmp views

Name          OID Tree          Type
-----
Default      iso              Included
Default      snmpNotificationMIB Excluded
DefaultSuper iso              Included

```

## 10.17 show snmp groups

Use the **show snmp groups** Privileged EXEC mode command to display the configured SNMP groups.

### Syntax

```
show snmp groups [groupname]
```

### Parameters

**groupname**—Specifies the group name. (Length: 1–30 characters)

### Command Mode

Privileged EXEC mode

### Example

The following example displays the configured SNMP groups.

```
Console# show snmp groups
```

Name	Model	Security		Views	
		Level	Read	Write	Notify
user-group	V3	priv	Default	" "	" "
managers-gro	V3	priv	Default	Default	" "

up

The following table describes significant fields shown above.

Field	Description
<b>Name</b>	Group name.
<b>Security Model</b>	SNMP model in use (v1, v2 or v3).
<b>Security Level</b>	Packet authentication with encryption. Applicable to SNMP v3 security only.

Field	Description	
<b>Views</b>	Read	View name enabling viewing the agent contents. If unspecified, all objects except the community-table and SNMPv3 user and access tables are available.
	Write	View name enabling data entry and managing the agent contents.
	Notify	View name enabling specifying an inform or a trap.

## 10.18 show snmp filters

Use the **show snmp filters** Privileged EXEC mode command to display the configured SNMP filters.

### Syntax

**show snmp filters** [*filtername*]

### Parameters

**filtername**—Specifies the filter name. (Length: 1–30 characters)

### Command Mode

Privileged EXEC mode

### Example

The following example displays the configured SNMP filters.

```
Console# show snmp filters
```

```

Name                OID Tree                Type
-----
user-filter         1.3.6.1.2.1.1          Included
user-filter         1.3.6.1.2.1.1.7       Excluded
user-filter         1.3.6.1.2.1.2.2.1.*.1  Included

```

---

## 10.19 show snmp users

Use the **show snmp users** Privileged EXEC mode command to display the configured SNMP users.

### Syntax

```
show snmp users [username]
```

### Parameters

**username**—Specifies the user name. (Length: 1–30 characters)

### Command Mode

Privileged EXEC mode

### Example

The following example displays the configured SNMP users.

---

```
Console# show snmp users
```

Name	Group name	Auth Method	Remote
John	user-group	md5	
John	user-group	md5	08009009020C0B099C07-5879

---

---

# 11 Web Server Commands

---

## 11.1 ip http server

The **ip http server** Global Configuration mode command enables configuring and monitoring the device from a web browser. Use the **no** form of this command to disable this function.

### Syntax

**ip http server**

**no ip http server**

### Default Configuration

HTTP server is enabled.

### Command Mode

Global Configuration mode

### Example

The following example enables configuring the device from a web browser.

---

```
Console(config)# ip http server
```

---

## 11.2 ip http timeout-policy

Use the **ip http timeout-policy** Global Configuration mode command to set the interval for the system to wait for user input in http/https sessions before automatic logoff. Use the **no** form of this command to return to the default value.

### Syntax

**ip http timeout-policy *idle seconds* [http-only | https-only]**

**no ip http timeout-policy**

### Parameters

**seconds**—Specifies the maximum number of seconds that a connection is kept open if no data is received or response data cannot be sent out. (Range: 0–86400)

**http-only** —The timeout is specified only for http

**https-only**— The timeout is specified only for https

### Default Configuration

600 seconds

### Command Mode

Global Configuration mode

### User Guidelines

This command also configures the timeout-policy for HTTPS.

To specify no timeout, enter the **ip http timeout-policy 0** command.

### Example

The following example configures the http timeout to be 1000 seconds.

---

```
Console(config)# ip http timeout-policy 1000
```

---

## 11.3 ip http secure-server

Use the **ip http secure-server** Global Configuration mode command to enable the device to be configured securely from a browser, and to also enable the device to be monitored or have its configuration modified securely from a browser. Use the **no** form of this command to disable this function.

### Syntax

**ip http secure-server**

**no ip http secure-server**

### Parameters

N/A

### Default Configuration

Disabled

### Command Mode

Global Configuration mode

### User Guidelines

Use the **crypto certificate generate** command to generate an HTTPS certificate.

### Example

---

```
console(config)# ip http secure-server
```

---

## 11.4 ip https certificate

The **ip https certificate** Global Configuration mode command configures the active certificate for HTTPS. Use the **no** form of this command to restore the default configuration.

### Syntax

**ip https certificate** *number*

**no ip https certificate**

### Parameters

**number**—Specifies the certificate number. (Range: 1–2)

### Default Configuration

The default certificate number is 1.

### Command Mode

Global Configuration mode

### User Guidelines

Use the **crypto certificate generate** command to generate a HTTPS certificate.

### Example

---

The following example configures the active certificate for HTTPS.

---



---

```
Console(config)# ip https certificate 2
```

---

---

## 11.5 show ip http

The **show ip http** EXEC mode command displays the HTTP server configuration.

### Syntax

**show ip http**

### Command Mode

EXEC mode

### Example

The following example displays the HTTP server configuration.

---

```
Console# show ip http
HTTP server enabled
Port: 80
Interactive timeout: 10 minutes
```

---

---

## 11.6 show ip https

The **show ip https** Privileged EXEC mode command displays the HTTPS server configuration.

### Syntax

**show ip https**

### Command Mode

Privileged EXEC mode

### Example

The following example displays the HTTPS server configuration.

---

```
Console# show ip https
```

```
HTTPS server enabled
```

```
Port: 443
```

```
Interactive timeout: Follows the HTTP interactive timeout (10 minutes)
```

```
Certificate 1 is active
```

```
Issued by: www.verisign.com
```

```
Valid from: 8/9/2003 to 8/9/2004
```

```
Subject: CN= router.gm.com, O= General Motors, C= US
```

```
Finger print: DC789788 DC88A988 127897BC BB789788
```

```
Certificate 2 is inactive
```

```
Issued by: self-signed
```

```
Valid from: 8/9/2003 to 8/9/2004
```

```
Subject: CN= router.gm.com, O= General Motors, C= US
```

```
Finger print: 1873B936 88DC3411 BC8932EF 782134BA
```

# 12 Teletype Network (Telnet), Secure Shell (SSH) and Secure Login (Slogin) Commands

---

## 12.1 ip telnet server

The **ip telnet server** Global Configuration mode command enables the device to be configured from a Telnet server. Use the **no** form of this command to disable the device configuration from a Telnet server.

### Syntax

**ip telnet server**

**no ip telnet server**

### Default Configuration

Device configuration from a Telnet server is disabled by default.

### Command Mode

Global Configuration mode

### User Guidelines

To control the device configuration by SSH, use the **ip ssh server** Global Configuration mode command.

### Example

The following example enables the device to be configured from a Telnet server.

---

```
Console(config)# ip telnet server
```

---

## 12.2 ip ssh server

The **ip ssh server** Global Configuration mode command enables the device to be configured from an SSH server. Use the **no** form of this command to disable the device configuration from a SSH server.

### Syntax

**ip ssh server**

## no ip ssh server

### Default Configuration

Device configuration from an SSH server is disabled.

### Command Mode

Global Configuration mode

### User Guidelines

If encryption keys are not generated, the SSH server is in standby until the keys are generated. To generate SSH server keys, use the **crypto key generate dsa** and **crypto key generate rsa** Global Configuration mode commands.

### Example

The following example enables configuring the device from a SSH server.

---

```
Console(config)# ip ssh server
```

---

## 12.3 user-key

The **user-key** SSH Public Key-string Configuration mode command specifies which SSH public key is manually configured. Use the **no** form of this command to remove an SSH public key.

### Syntax

**user-key** *username* {*rsa* | *dsa*}

**no user-key** *username*

### Parameters

- **username**—Specifies the remote SSH client username. (Length: 1–48 characters)
- **rsa**—Specifies that the RSA key pair is manually configured.
- **dsa**—Specifies that the DSA key pair is manually configured.

### Default Configuration

No SSH public keys exist.

## Command Mode

SSH Public Key-string Configuration mode

## User Guidelines

Follow this command with the **key-string** SSH Public Key-String Configuration mode command to specify the key.

Please note that after entering this command, the existing key is deleted even if no new key is defined by the **key-string** command

## Example

The following example enables manually configuring an SSH public key for SSH public key-chain **bob**.

---

```
Console(config)# crypto key pubkey-chain ssh
Console(config-pubkey-chain)# user-key bob rsa
Console(config-pubkey-key)# key-string row
AAAAB3NzaC1yc2EAAAADAQABAAQACvTnRwPW1
```

---

## 12.4 key-string

The **key-string** SSH Public Key-string Configuration mode command manually specifies an SSH public key.

### Syntax

**key-string** [*row key-string*]

### Parameters

- **row**—Specifies the SSH public key row by row.
- **key-string**—Specifies the key in UU-encoded DER format. UU-encoded DER format is the same format as in the `authorized_keys` file used by OpenSSH. (Length:0–160)

### Default Configuration

Keys do not exist.

## Command Mode

### SSH Public Key-string Configuration mode

#### User Guidelines

Use the **key-string** SSH Public Key-string Configuration mode command without the **row** parameter to specify which SSH public key is to be interactively configured next. Enter a row with no characters to complete the command.

Use the **key-string row** SSH Public Key-string Configuration mode command to specify the SSH public key, row by row. Each row must begin with a **key-string row** command.

The UU-encoded DER format is the same format as in the `authorized_keys` file used by OpenSSH.

#### Example

The following example enters public key strings for SSH public key client 'bob'.

---

```

Console(config)# crypto key pubkey-chain ssh
Console(config-pubkey-chain)# user-key bob rsa
Console(config-pubkey-key)# key-string
AAAAB3NzaC1yc2EAAAADAQABAAQACvTnRwPWl
Al4kpgIw9GBRonZQZxjHKcqKL6rMlQ+
ZNXfZSkvHG+QusIZ/76ILmFT34v7u7ChFAE+
Vu4GRfpSwoQUvV35LqJJK67IOU/zfwO1lg
kTwm175QR9gHujS6KwGN2QWXgh3ub8gDjTSq
muSn/Wd05iDX2IExQWu08licg1k02LYciz
+Z4TrEU/9FJxwPiVQOjc+KBXuR0juNg5nFYsY
0ZCk0N/W9a/tnkmlshRE7Di7l+w3fNiOA
6w9o44t6+AINEICCCA4YcF6zMzaTlwefWwX6f+
Rmt5nhhqAtN/4oJfcel66DqVXlgWmN
zNR4DYDvSzg0lDnwCAC8Qh

Fingerprint: a4:16:46:23:5a:8d:1d:b5:37:59:eb:44:13:b9:33:e9

```

```
Console(config)# crypto key pubkey-chain ssh
Console(config-pubkey-chain)# user-key bob rsa
Console(config-pubkey-key)# key-string row AAAAB3Nza
Console(config-pubkey-key)# key-string row C1yc2
```

---

## 12.5 crypto key pubkey-chain ssh

The **crypto key pubkey-chain ssh** Global Configuration mode command enters the SSH Public Key-chain Configuration mode. This mode is used to manually specify other device public keys such as SSH client public keys.

### Syntax

**crypto key pubkey-chain ssh**

### Default Configuration

Keys do not exist.

### Command Mode

Global Configuration mode

### User Guidelines

Use this command when you want to manually specify SSH client's public keys.

### Example

The following example enters the SSH Public Key-chain Configuration mode and manually configures the RSA key pair for SSH public key-chain to 'bob'.

---

```
Console(config)# crypto key pubkey-chain ssh
Console(config-pubkey-chain)# user-key bob
Console(config-pubkey-key)# key-string rsa
AAAAB3NzaC1yc2EAAAADAQABAAQ=CvTnRwPW1
A14kpbqIw9GBRonZQZxjHKcckL6rMlQ+
ZNXfZSkvHG+QusIZ/76ILmFT34v7u7ChFAE+
```

```
Vu4GRfpSwoQUvV35LqJJK67IOU/zfw011g
kTwm175QR9gHujS6KwGN2QWXgh3ub8gDjTSq
muSn/Wd05iDX2IExQWu08licg1k02LYciz
+Z4TrEU/9FJxwPiVQOjc+KBXuR0juNg5nFYsY
0ZCk0N/W9a/tnkmlshRE7Di71+w3fNiOA
6w9o44t6+AINEICCCA4YcF6zMzaTlwefWwX6f+
Rmt5nhhqAtN/4oJfce166DqVXlgWmN
zNR4DYDvSzg01DnwCAC8Qh
```

```
Fingerprint: a4:16:46:23:5a:8d:1d:b5:37:59:eb:44:13:b9:33:e9
```

---

## 12.6 show crypto key pubkey-chain ssh

The **show crypto key pubkey-chain ssh** Privileged EXEC mode command displays SSH public keys stored on the device.

### Syntax

```
show crypto key pubkey-chain ssh [username username] [fingerprint  
{bubble-babble | hex}]
```

### Parameters

- **username *username***—Specifies the remote SSH client username. (Length: 1–48 characters)
- **fingerprint {*bubble-babble* | *hex*}**—Specifies the fingerprint display format. The possible values are:
  - bubble-babble***—Specifies that the fingerprint is displayed in Bubble Babble format.
  - hex***—Specifies that the fingerprint is displayed in hexadecimal format.

### Default Configuration

The default fingerprint format is hexadecimal.



## Command Mode

Privileged EXEC mode

## Example

The following examples display SSH public keys stored on the device.

---

```
Console# show crypto key pubkey-chain ssh
```

```
Username
```

```
-----
```

```
bob
```

```
john
```

```
Fingerprint
```

```
-----
```

```
9A:CC:01:C5:78:39:27:86:79:CC:23:C5:98:59:F1:86
```

```
98:F7:6E:28:F2:79:87:C8:18:F8:88:CC:F8:89:87:C8
```

```
Console# show crypto key pubkey-chain ssh username bob
```

```
Username: bob
```

```
Key: 005C300D 06092A86 4886F70D 01010105 00034B00 30480241 00C5E23B 55D6AB22  
04AEF1BA A54028A6 9ACC01C5 129D99E4
```

```
Fingerprint: 9A:CC:01:C5:78:39:27:86:79:CC:23:C5:98:59:F1:86
```

---

## 12.7 show ip ssh

The **show ip ssh** Privileged EXEC mode command displays the SSH server configuration.

### Syntax

**show ip ssh**

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays the SSH server configuration.

```

Console# show ip ssh

SSH server enabled. Port: 22

RSA key was generated.

DSA (DSS) key was generated.

SSH Public Key Authentication is enabled.

Active incoming sessions:
IP address   SSH username   Version       Cipher        Auth code
-----
172.16.0.1   John Brown     1.5          3DES         HMAC-SHA1

```

The following table describes the significant fields shown in the display.

Field	Description
<b>IP address</b>	The client address
<b>SSH username</b>	The user name
<b>Version</b>	The SSH version number
<b>Cipher</b>	The encryption type (3DES, Blowfish, RC4)
<b>Auth Code</b>	The authentication Code (HMAC-MD5, HMAC-SHA1)

# 13 Line Commands

---

## 13.1 line

The **line** Global Configuration mode command identifies a specific line for configuration and enters the Line Configuration command mode.

### Syntax

**line** {*console* / *telnet* / *ssh*}

### Parameters

- **console**—Enters the console terminal line mode.
- **telnet**—Configures the device as a virtual terminal for remote console access (Telnet).
- **ssh**—Configures the device as a virtual terminal for secured remote console access (SSH).

### Command Mode

Global Configuration mode

### Example

The following example configures the device as a virtual terminal for remote (Telnet) console access.

---

```
Console(config)# line telnet
```

```
Console(config-line)#
```

---

## 13.2 speed

The **speed** Line Configuration mode command sets the line baud rate. Use the **no** form of this command to restore the default configuration.

### Syntax

**speed** *bps*

**no speed**

### Parameters

**bps**—Specifies the baud rate in bits per second (bps). Possible values are 2400, 4800, 9600, 19200, 38400, 57600, and 115200.

### Default Configuration

The default speed is 115200 bps.

### Command Mode

Line Configuration (console) mode

### User Guidelines

The configured speed is applied when Autobaud is disabled. This configuration applies to the current session only.

### Example

The following example configures the line baud rate as 9600 bits per second.

---

```
Console(config-line)# speed 9600
```

---

## 13.3 autobaud

The **autobaud** Line Configuration mode command sets the line for automatic baud rate detection (autobaud). Use the **no** form of this command to disable automatic baud rate detection.

### Syntax

**autobaud**

**no autobaud**

### Default Configuration

Automatic baud rate detection is enabled.

### Command Mode

Line Configuration mode

### User Guidelines

To start communication using Autobaud, press the **Enter** key twice.

### Example

The following example enables autobaud.

---

```
Console(config)# line console
Console(config-line)# autobaud
```

---

## 13.4 exec-timeout

The **exec-timeout** Line Configuration mode command sets the session idle time interval, during which the system waits for user input before automatic logoff. Use the **no** form of this command to restore the default configuration.

### Syntax

**exec-timeout** *minutes* [*seconds*]

**no exec-timeout**

### Parameters

- **minutes**—Specifies the number of minutes. (Range: 0-65535)
- **seconds**—Specifies the number of seconds. (Range: 0-59)

### Default Configuration

The default idle time interval is 10 minutes.

### Command Mode

Line Configuration mode

### Example

The following example sets the HTTP session idle time interval before automatic logoff to 20 minutes and 10 seconds.

---

```
Console(config)# line console
Console(config-line)# exec-timeout 20 10
```

---

## 13.5 show line

The **show line** EXEC mode command displays line parameters.

### Syntax

**show line** [*console / telnet / ssh*]

### Parameters

- **console**—Displays the console configuration.
- **telnet**—Displays the Telnet configuration.
- **ssh**—Displays the SSH configuration.

### Default Configuration

If the line is not specified, all line configuration parameters are displayed.

### Command Mode

EXEC mode

### Example

The following example displays the line configuration.

---

```
Console> show line
Console configuration:
Interactive timeout: Disabled
History: 10
Baudrate: 9600
Databits: 8
Parity: none
Stopbits: 1

Telnet configuration:
Telnet is enabled.
```

---

```
Interactive timeout: 10 minutes 10 seconds
```

```
History: 10
```

```
SSH configuration:
```

```
SSH is enabled.
```

```
Interactive timeout: 10 minutes 10 seconds
```

```
History: 10
```

# 14 Bonjour Commands (Cisco Only)

**Table 1: Internal Revision History**

Date	By	Change
25-Jan-2010	Ava	New commands from Ilan Yerushalmi (Commands for 750 - file 4.doc)
7-Mar-10	Ilan	The syntax of "bonjour mode interface range" command was updated
June 22	Patricia	Minor editorial and formatting changes
July 12	Patricia	Removed Non-ISCLI text. Note that there are commands without examples.
31-Jan-11	Michael	The example of <b>bonjour interface range</b> was modified.

## 14.1 bonjour enable

Date	TW Status	Release	Modification
		750	The command was first introduced

Use the **bonjour enable** Global Configuration mode command to enable Bonjour globally. Use the **no** format of the command to disable globally.

### Syntax

**bonjour enable**

**no bonjour enable.**

### Default Configuration

Enable

### Command Mode

Global Configuration mode

### Examples



```
console(conf)# bonjour enable
```

## 14.2 bonjour interface range

Date	TW Status	Release	Modification
		750	The command was first introduced
31-Jan-11	Michael		
7 Feb 11	Gina	750	Add user guidelines

Use the **bonjour interface range** Global Configuration mode command to add L2 interfaces to the Bonjour L2 Interface List. Use the **no** format of the command to remove L2 interfaces from the list.

### Syntax

```
bonjour interface range {interface-list}
```

### Parameters

**interface-list**—Specifies a list of interfaces, which can be of the following types:

- Ethernet port
- Port-channel
- VLAN

### Default Configuration

The list is empty.

### Command Mode

Global Configuration mode

### User Guidelines

This command can only be used if the device is in Layer 3 (router) mode.

### Examples

```
console(config)# bonjour interface range gil-3
```

## 14.3 show bonjour

Date	TW Status	Release	Modification
		750	The command was first introduced
7-Feb-11	Gina	750	New examples

Use the **show bonjour** Privileged EXEC mode command to show Bonjour information

### Syntax

**show bonjour** [*interface-id*]

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types:

- Ethernet port
- Port-channel
- VLAN

### Command Mode

Privileged EXEC mode

### Examples

#### Layer 2:

```
console# show bonjour
```

```
Bonjour status: enabled
```

```
L2 interface status: Up
```

```
IP Address: 10.5.226.46
```

```
Service   Admin Status   Oper Status
```

```
-----
```

```
csc0-sb   enabled       enabled
```

http	enabled	enabled
https	enabled	disabled
ssh	enabled	disabled
telnet	enabled	disabled

**Layer 3:**

```
console# show bonjour
```

```
Bonjour global status: enabled
```

```
Bonjour L2 interfaces port list: vlans 1
```

Service	Admin Status	Oper Status
-----	-----	-----
cisco-sb	enabled	enabled
http	enabled	enabled
https	enabled	disabled
ssh	enabled	disabled
telnet	enabled	disabled

# 15 Authentication, Authorization and Accounting (AAA) Commands

## 15.1 aaa authentication login

The **aaa authentication login** Global Configuration mode command sets one or more authentication methods to be applied during login. The list of authentication methods may be assigned a list name, and this list name can be used in **aaa authentication login** and **aaa authentication enable**. Use the **no** form of this command to restore the default authentication method.

### Syntax

```
aaa authentication login {default / list-name} method1 [method2...]
```

```
aaa authentication login list-name method1 method2...
```

```
no aaa authentication login {default / list-name}
```

### Parameters

- **default**—Uses the authentication methods that follow this argument as the default method list when a user logs in.
- **list-name**—Specifies a name of a list of authentication methods activated when a user logs in. (Length: 1–12 characters)
- **method1 [method2...]**—Specifies a list of methods that the authentication algorithm tries (in the given sequence). The additional authentication methods are used only if the previous method returns an error, not if it fails. To ensure that the authentication succeeds even if all methods return an error, specify **none** as the final method in the command line. Select one or more methods from the following list::

Keyword	Description
<b>enable</b>	Uses the enable password for authentication.
<b>line</b>	Uses the line password for authentication.
<b>local</b>	Uses the local username database for authentication.
<b>none</b>	Uses no authentication.
<b>radius</b>	Uses the list of all RADIUS servers for authentication.
<b>tacacs</b>	Uses the list of all TACACS+ servers for authentication.

### Default Configuration

The local user database is the default authentication method. This is the same as entering the command **aaa authentication login local**.

**NOTE** If an authentication method is not defined, console users can log in without any authentication verification.

### Command Mode

Global Configuration mode

### User Guidelines

Create a list of authentication methods by entering the **aaa authentication login *list-name method1 [method2...]***, where *list-name* is any character string used to name this list. The method arguments identifies the list of methods that the authentication algorithm tries, in the given sequence.

The default and list names created with this command are used with **aaa authentication login** and **aaa authentication enable**.

The additional methods of authentication are used only if the previous method returns an error, not if it fails. Specify **none** as the final method in the command line to ensure that the authentication succeeds even if all methods return an error.

**no aaa authentication login *list-name*** deletes *list-name* if it has not been referenced.

### Example

The following example sets the authentication login methods for the console.

```
console (config)# aaa authentication login authen-list radius local none
console (config)#line console
console (config-line)#login authentication authen-list
```

## 15.2 aaa authentication enable

The **aaa authentication enable** Global Configuration mode command sets an authentication method for accessing higher privilege levels. A user, who logons with a lower privilege level, must enter this password to access a higher level.

To restore the default authentication method, use the **no** form of this command.

## Syntax

**aaa authentication enable** *{default / list-name}* *method* [*method2...*]

**no aaa authentication enable** *{default / list-name}*

## Parameters

- **default**—Uses the listed authentication methods that follow this argument as the default method list, when accessing higher privilege levels.
- **list-name** —Specifies a name for the list of authentication methods activated when a user accesses higher privilege levels. (Length: 1–12 characters)
- **method [method2...]**—Specifies a list of methods that the authentication algorithm tries, in the given sequence. The additional authentication methods are used only if the previous method returns an error, not if it fails. Specify **none** as the final method in the command line to ensure that the authentication succeeds, even if all methods return an error. Select one or more methods from the following list:

Keyword	Description
<b>enable</b>	Uses the enable password for authentication.
<b>line</b>	Uses the line password for authentication.
<b>none</b>	Uses no authentication.
<b>radius</b>	Uses the list of all RADIUS servers for authentication. Uses username "\$enabx\$." where x is the privilege level.
<b>tacacs</b>	Uses the list of all TACACS servers for authentication. Uses username "\$enabx\$." where x is the privilege level.

## Default Configuration

The **enable password** command is the default authentication login method. This is the same as entering the command **aaa authentication enable default enable**.

On a console, the enable password is used if a password exists. If no password is set, authentication still succeeds. This is the same as entering the command **aaa authentication enable default enable none**.

## Command Mode

Global Configuration mode

### User Guidelines

Create a list by entering the **aaa authentication enable** *list-name method1 [method2...]* command where *list-name* is any character string used to name this list. The method argument identifies the list of methods that the authentication algorithm tries, in the given sequence.

The default and list names created with the **aaa authentication enable** command are used with **aaa authentication login** and **aaa authentication enable**.

All **aaa authentication enable default** requests sent by the device to a RADIUS or TACACS+ server include the username \$enabx\$, where *x* is the requested privilege level.

The additional methods of authentication are used only if the previous method returns an error, not if it fails. Specify **none** as the final method in the command line to ensure that the authentication succeeds even if all methods return an error.

**no aaa authentication enable** *list-name* deletes *list-name* if it has not been referenced.

### Example

The following example sets the enable password for authentication for accessing higher privilege levels.

```
Console(config)# aaa authentication enable enable-list radius none
Console(config)# line console
Console(config-line)# enable authentication enable-list
```

## 15.3 ip http authentication

The **ip http authentication** Global Configuration mode command specifies authentication methods for HTTP server access. Use the **no** form of this command to restore the default authentication method.

### Syntax

**ip http authentication aaa login-authentication** *method1 [method2...]*

**no ip http authentication aaa login-authentication**

### Parameters

**method [method2...]**—Specifies a list of methods that the authentication algorithm tries, in the given sequence. The additional authentication methods are used only if the previous method returns an error, not if it fails. Specify **none** as the final method in the command line to ensure that the authentication succeeds, even if all methods return an error. Select one or more methods from the following list:

Keyword	Description
<b>local</b>	Uses the local username database for authentication.
<b>none</b>	Uses no authentication.
<b>radius</b>	Uses the list of all RADIUS servers for authentication.
<b>tacacs</b>	Uses the list of all TACACS+ servers for authentication.

### Default Configuration

The local user database is the default authentication login method. This is the same as entering the **ip http authentication local** command.

### Command Mode

Global Configuration mode

### User Guidelines

The command is relevant for HTTP and HTTPS server users.

The additional methods of authentication are used only if the previous method returns an error, not if it fails. Specify **none** as the final method in the command line to ensure that the authentication succeeds, even if all methods return an error.

### Example

The following example specifies the HTTP access authentication methods.

```
Console(config)# ip http authentication aaa login-authentication radius local
```

## 15.4 show authentication methods

The **show authentication methods** Privileged EXEC mode command displays information about the authentication methods.



**Syntax****show authentication methods****Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays the authentication configuration.

---

```
Console# show authentication methods
```

```
Login Authentication Method Lists
```

```
-----
```

```
Default: Radius, Local, Line
```

```
Console_Login: Line, None
```

```
Enable Authentication Method Lists
```

```
-----
```

```
Default: Radius, Enable
```

```
Console_Enable: Enable, None
```

Line -----	Login Method List	Enable Method List -----
Console	-----	Console_Enable
Telnet	Console_Login	Default
SSH	Default	Default
	Default	
HTTP: Radius, local		
HTTPS: Radius, local		
Dot1x: Radius		

## 15.5 password

The **password** Line Configuration mode command specifies a password on a line, also known as access method, such as a console or Telnet. Use the **no** form of this command to return to the default password.

### Syntax

**password** *password* [*encrypted*]

**no password**

### Parameters

- **password**—Specifies the password for this line. (Length: 0–159 characters)
- **encrypted**—Specifies that the password is encrypted and copied from another device configuration.

### Default Configuration

No password is defined.

### Command Mode

Line Configuration mode

### Example

The following example specifies the password 'secret' on a console.

```
Console(config)# line console
Console(config-line)# password secret
```

## 15.6 enable password

Use the **enable password** Global Configuration mode command to set a local password to control access to normal and privilege levels. Use the **no** form of this command to return to the default password.

When the administrator configures a new **enable** password, this password is encrypted automatically and saved to the configuration file. No matter how the password was entered, it appears in the configuration file with the keyword **encrypted** and the encrypted value.

If the administrator wants to manually copy a password that was configured on one switch (for instance, switch B) to another switch (for instance, switch A), the administrator must add **encrypted** in front of this encrypted password when entering the **enable** command in switch A. In this way, the two switches will have the same password.

### Syntax

```
enable password [level privilege-level] {unencrypted-password | encrypted encrypted-password}
```

```
no enable password [level level]
```

### Parameters

- **level** *privilege-level*—Level for which the password applies. If not specified the level is 15. (Range: 1–15)
- **password** *unencrypted-password*—Password for this level. (Range: 0–159 chars)
- **password encrypted** *encrypted-password*—Specifies that the password is encrypted. Use this keyword to enter a password that is already encrypted (for instance that you copied from another the configuration file of another device). (Range: 1–40)

### Default Configuration

Default for **level** is 15.

Passwords are encrypted by default.

### Command Mode

Global Configuration mode

### User Guidelines

Passwords are encrypted by default. You only are required to use the **encrypted** keyword when you are actually entering an encrypted keyword.

### Example

The first command sets an unencrypted password for level 7 (it will be encrypted in the configuration file).

The second command sets a password that has already been encrypted. It will be copied to the configuration file just as it is entered. To use it, the user must know its unencrypted form.

---

```
console(config)# enable password level 7 let-me-in
console(config)# enable password level 15 encrypted
4b529f21c93d4706090285b0c10172eb073ffe4
```

---

## 15.7 username

Use the **username** Global Configuration mode command to establish a username-based authentication system. Use the **no** form to remove a user name.

### Syntax

**username** *name* {*nopassword* | *password password* | *privilege privilege-level* | *unencrypted-password* | *encrypted encrypted-password*}

**username** *name*

**no username** *name*

### Parameters

- **name**—The name of the user. (Range: 1–20 characters)

- **nopassword**—No password is required for this user to log in.
- **unencrypted-password**—The authentication password for the user. (Range: 1–159)
- ***encrypted encrypted-password***—Specifies that the password is encrypted. Use this keyword to enter a password that is already encrypted (for instance that you copied from another the configuration file of another device). (Range: 1–40)
- **privilege *privilege-level***—Privilege level for which the password applies. If not specified the level is 15. (Range: 1–15).

### Default Configuration

No user is defined.

### Command Mode

Global Configuration mode

### Usage Guidelines

See "[User \(Privilege\) Levels](#)" for an explanation of privilege levels.

### Example

The first command sets an unencrypted password for user tom (it will be encrypted in the configuration file).

The second command sets a password for user jerry that has already been encrypted. It will copied to the configuration file just as it is entered. To use it, the user must know its unencrypted form.

---

```
console(config)# username tom privilege 15 password 1234
console(config)# username jerry privilege 15 encrypted
4b529f21c93d4706090285b0c10172eb073ffebc4
```

---

## 15.8 show user accounts

The **show user accounts** Privileged EXEC mode command displays information about the users local database.

**Syntax****show user accounts****Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays information about the users local database.

---

```
Console# show user accounts
```

```

Username      Privilege
-----
Bob           15
Robert        15
Smith         15

```

The following table describes the significant fields shown in the display:

Field	Description
<b>Username</b>	The user name.
<b>Privilege</b>	The user's privilege level.

---

## 15.9 passwords complexity enable

Use the **passwords complexity enable** Global Configuration mode command to enforce minimum password complexity. The **no** form of this command disables enforcing password complexity.

**Syntax****passwords complexity enable****no passwords complexity enable****Parameters****Parameters**

N/A

**Default Configuration**

Enabled

**Command Mode**

Global Configuration mode

**User Guidelines**

If password complexity is enabled **by default**, the user is forced to enter a password that:

- Has a minimum length of 8 characters.
- Contains characters from at least 3 character classes (uppercase letters, lowercase letters, numbers, and special characters available on a standard keyboard).
- Are different from the current password.
- Contains no character that is repeated more than 3 times consecutively.
- Does not repeat or reverse the user name or any variant reached by changing the case of the characters.
- Does not repeat or reverse the manufacturer's name or any variant reached by changing the case of the characters.

You can control the above attributes of password complexity with specific commands described in this section.

If you have previously configured other complexity settings, then those settings are used. This command does not wipe out the other settings. It works only as a toggle.

### Example

The following example configures requiring complex passwords that fulfill the minimum requirements specified in the User Guidelines above.

```
console(config)# passwords complexity enable

console#show passwords configuration

Passwords aging is enabled with aging time 180 days.

Passwords complexity is enabled with the following attributes:

Minimal length: 3 characters

Minimal classes: 3

New password must be different than the current: Enabled

Maximum consecutive same characters: 3

New password must be different than the user name: Enabled

New password must be different than the manufacturer name: Enabled

switchcc293e#
```

## 15.10 passwords complexity <attributes>

Use the **passwords complexity <attributes>** Global Configuration mode commands to control the minimum requirements from a password when password complexity is enabled. Use the **no** form of these commands to return to default.

### Syntax

**passwords complexity *min-length* number**

**no passwords complexity *min-length***

**passwords complexity *min-classes* number**

**no passwords complexity *min-classes***

**passwords complexity *not-current***

**no passwords complexity *not-current***

**passwords complexity *no-repeat* number**

**no password complexity *no-repeat***



**passwords complexity *not-username***

**no passwords complexity *not-username***

**passwords complexity *not-manufacturer-name***

**no passwords complexity *not-manufacturer-name***

### Parameters

- **min-length number**—Sets the minimal length of the password. (Range: 0–64)
- **min-classes number**—Sets the minimal character classes (uppercase letters, lowercase letters, numbers, and special characters available on a standard keyboard). (Range: 0–4)
- **not-current**—Specifies that the new password cannot be the same as the current password.
- **no-repeat number**—Specifies the maximum number of characters in the new password that can be repeated consecutively. Zero specifies that there is no limit on repeated characters. (Range: 0–16)
- **not-username**—Specifies that the password cannot repeat or reverse the user name or any variant reached by changing the case of the characters.
- **not-manufacturer-name**—Specifies that the password cannot repeat or reverse the manufacturer's name or any variant reached by changing the case of the characters.

### Default Configuration

The minimal length is 8.

The number of classes is 3.

The default for no-repeat is 3.

All the other controls are enabled by default.

### Command Mode

Global Configuration mode

### Example

The following example configures the minimal required password length to 8 characters.

---

```
Console (config)# passwords complexity min-length 8
```

---

## 15.11 passwords aging

Use the **passwords aging** Global Configuration mode command to enforce password aging. Use the **no** form of this command to return to default.

### Syntax

**passwords aging** *days*

**no passwords aging**

### Parameters

**days**—Specifies the number of days before a password change is forced. You can use 0 to disable aging. (Range: 0–365)

### Default Configuration

Enabled and the number of days is 180 days

### Command Mode

Global Configuration mode

### User Guidelines

Aging is relevant only to users of the local database with privilege level 15 and to “enable” a password of privilege level 15.

To disable password aging, use **passwords aging 0**.

### Example

The following example configures the aging time to be 24 days.

---

```
Console (config)# passwords aging 24
```

---

## 15.12 show passwords configuration

The **show passwords configuration** Privileged EXEC mode command displays information about the password management configuration.

### Syntax

**show passwords configuration**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### Example

---

```
console#show passwords configuration
Passwords aging is enabled with aging time 180 days.
Passwords complexity is enabled with the following attributes:
Minimal length: 3 characters
Minimal classes: 3
New password must be different than the current: Enabled
Maximum consecutive same characters: 3
New password must be different than the user name: Enabled
New password must be different than the manufacturer name: Enabled
switchcc293e#
```

The following table describes the significant fields shown in the display:

Field	Description
<b>Minimal length</b>	The minimal length required for passwords in the local database.
<b>Minimal character classes</b>	The minimal number of different types of characters (special characters, integers and so on) required to be part of the password.
<b>Maximum number of repeated characters</b>	The maximum number of times a single character can be repeated in the password.
<b>Level</b>	The applied password privilege level.
<b>Aging</b>	The password aging time in days.

# 16 Remote Authentication Dial-In User Service (RADIUS) Commands

## 16.1 radius-server host

Use the **radius-server host** Global Configuration mode command to specify a RADIUS server host. Use the no form of the command to delete the specified RADIUS server host.

### Syntax

```
radius-server host {ipv4-address | ipv6-address | ipv6z-address | hostname}
[auth-port auth-port-number] [timeout timeout] [retransmit retries] [deadtime
deadtime] [key key-string] [source {ipv4-address | ipv6-address}] [priority priority]
[usage {login | 802.1x | all}]
```

```
no radius-server host {ipv4-address | ipv6-address | hostname}
```

### Parameters

- **ipv4-address**—Specifies the RADIUS server host IPv4 address.
- **ipv6-address**—Specifies the RADIUS server host IPv6 address.
- **ipv6z-address**—Specifies the RADIUS server host IPv6Z address. The IPv6Z address format is: **{ipv6-link-local-address}%{interface-name}**. The subparameters are:
  - **ipv6-link-local-address**—Specifies the IPv6 Link Local address.
  - **interface-name**—Specifies the outgoing interface name. The interface name has the format: **vlan{integer} | po{integer} | isatap{integer} | {physical-port-name}** If the interface-name is 0, this indicates the default interface.
  - The subparameter **integer** has the format: {decimal-digit} | {integer}{decimal-digit}. decimal-digit has the range 0–9.

The following combinations are possible:

- **ipv6\_address%interface\_id** - Refers to the IPv6 address on the interface specified.
- **ipv6\_address%0** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.

- **ipv6\_address** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **hostname**—Specifies the RADIUS server host name. Translation to IPv4 addresses only is supported. (Length: 1–158 characters. Maximum label length: 63 characters)
- **auth-port auth-port-number**—Specifies the port number for authentication requests. If the port number is set to 0, the host is not used for authentication. (Range: 0–65535)
- **timeout timeout**—Specifies the timeout value in seconds. (Range: 1–30)
- **retransmit retries**—Specifies the retransmit value. (Range: 1–10)
- **deadtime deadtime**—Specifies the length of time in minutes during which a RADIUS server is skipped over by transaction requests. (Range: 0–2000)
- **key key-string**—Specifies the authentication and encryption key for all RADIUS communications between the device and the RADIUS server. This key must match the encryption used on the RADIUS daemon. To specify an empty string, enter "". (Length: 0–128 characters)
- **source {ipv4-address | ipv6-address}**—Specifies the source IPv4 or IPv6 address to use for communication. 0.0.0.0 is interpreted as a request to use the IP address of the outgoing IP interface.
- **priority priority**—Specifies the order in which servers are used, where 0 has the highest priority. (Range: 0–65535)
- **usage {login | 802.1x | all}**—Specifies the RADIUS server usage type. The possible values are:
  - **login**—Specifies that the RADIUS server is used for user login parameters authentication.
  - **802.1x**—Specifies that the RADIUS server is used for 802.1x port authentication.
  - **all**—Specifies that the RADIUS server is used for user login parameters authentication and 802.1x port authentication.

### Default Configuration

No RADIUS host is specified; the global **radius-server** command values are the default values.

The default authentication port number is 1812.

If **timeout** is not specified, the global value is used.

If **retransmit** is not specified, the global value is used.

If **key-string** is not specified, the global value is used.

If the **source** value is not specified, the global value is used.

The default usage type is **all**.

### Command Mode

Global Configuration mode

### User Guidelines

To specify multiple hosts, multiple **radius-server host** commands can be used.

If no host-specific **timeout**, **retries**, **deadtime** or **key-string** values are specified, the global values apply to each RADIUS server host.

The **source** parameter address type must be the same as that of the **host** parameter.

### Example

The following example specifies a RADIUS server host with IP address 192.168.10.1, authentication request port number 20, and a 20-second timeout period.

```
Console(config)# radius-server host 192.168.10.1 auth-port 20 timeout 20
```

## 16.2 radius-server key

Use the **radius-server key** Global Configuration mode command to set the authentication and encryption key for all RADIUS communications between the device and the RADIUS daemon. Use the **no** form of this command to restore the default configuration.

### Syntax

**radius-server key** [*key-string*]

**no radius-server key**

### Parameters

**key-string**—Specifies the authentication and encryption key for all RADIUS communications between the device and the RADIUS server. This key must match the encryption used on the RADIUS daemon. (Range: 0–128 characters)

### Default Configuration

The key-string is an empty string.

### Command Mode

Global Configuration mode

### Example

The following example defines the authentication and encryption key for all RADIUS communications between the device and the RADIUS daemon.

---

```
Console(config)# radius-server key enterprise-server
```

---

## 16.3 radius-server retransmit

Use the **radius-server retransmit** Global Configuration mode command to specify the number of times the software searches the list of RADIUS server hosts. Use the no form of this command to restore the default configuration.

### Syntax

**radius-server retransmit** *retries*

**no radius-server retransmit**

### Parameters

**retransmit** *retries*—Specifies the retransmit value. (Range: 1–10)

### Default Configuration

The software searches the list of RADIUS server hosts 3 times.

### Command Mode

Global Configuration mode



### Example

The following example configures the number of times the software searches all RADIUS server hosts as 5.

```
console(config)# radius-server retransmit 5
```

## 16.4 radius-server source-ip

Use the **radius-server source-ip** Global Configuration mode command to specify the source IP address used for communication with RADIUS servers. Use the no form of this command to restore the default configuration.

### Syntax

```
radius-server source-ip {source}
```

```
no radius-server source-ip {source}
```

### Parameters

**source**—Specifies the source IP address.

### Default Configuration

The source IP address is the IP address of the outgoing IP interface.

### Command Mode

Global Configuration mode

### User Guidelines

If there is no available IP interface of the configured IP source address, an error message is issued when attempting to communicate with the IP address.

### Example

The following example configures the source IP address used for communication with all RADIUS servers to 10.1.1.1.

```
console(config)# radius-server source-ip 10.1.1.1
```

---

## 16.5 radius-server source-ipv6

Use the **radius-server source-ipv6** Global Configuration mode command to specify the source IPv6 address used for communication with RADIUS servers. Use the **no** form of this command to restore the default configuration.

### Syntax

**radius-server source-ipv6** {*source*}

**no radius-server source-ipv6** {*source*}

### Parameters

**source**—Specifies the source IPv6 address.

### Default Configuration

The source IP address is the IP address of the outgoing IP interface.

### Command Mode

Global Configuration mode

### User Guidelines

If there is no available IP interface of the configured IP source address, an error message is issued when attempting to communicate with the IP address.

### Example

The following example configures the source IP address used for communication with all RADIUS servers to 3ffe:1900:4545:3:200:f8ff:fe21:67cf.

---

```
console(config)# radius-server source-ipv6
3ffe:1900:4545:3:200:f8ff:fe21:67cf
```

---

## 16.6 radius-server timeout

Use the **radius-server timeout** Global Configuration mode command to set the time interval during which the device waits for a server host to reply. Use the **no** form of this command to restore the default configuration.

**Syntax**

**radius-server timeout** *timeout-seconds*

**no radius-server timeout**

**Parameters**

**timeout** *timeout-seconds*—Specifies the timeout value in seconds. (Range: 1–30)

**Default Configuration**

The default timeout value is 3 seconds.

**Command Mode**

Global Configuration mode

**Example**

The following example sets the timeout interval on all RADIUS servers to 5 seconds.

---

```
Console(config)# radius-server timeout 5
```

---

## 16.7 radius-server deadtime

Use the **radius-server deadtime** Global Configuration mode command to configure the time interval during which unavailable RADIUS servers are skipped over by transaction requests. This improves RADIUS response time when servers are unavailable. Use the **no** form of this command to restore the default configuration.

**Syntax**

**radius-server deadtime** *deadtime*

**no radius-server deadtime**

**Parameters**

**deadtime**—Specifies the time interval in minutes during which a RADIUS server is skipped over by transaction requests. (Range: 0–2000)

### Default Configuration

The default deadtime interval is 0.

### Command Mode

Global Configuration mode

### Example

The following example sets all RADIUS server deadtimes to 10 minutes.

---

```
Console(config)# radius-server deadtime 10
```

---

## 16.8 show radius-servers

Use the **show radius-servers** Privileged EXEC mode command to display the RADIUS server settings.

### Syntax

**show radius-servers**

### Command Mode

Privileged EXEC mode

**Example**

The following example displays RADIUS server settings.

```

Console# show radius-servers
          Port  Port  Time           Dead   Sourc
IP       Auth  Acct  Out           Retransmission  time   e      Priority  Usage
address  ----  ----  -----  -----  -----  IP    -----  -----
-----  1812  1813  Global  Global           Global  -----  1         All
172.16.1. 1812  1813  11      8              Global  Global  2         All
1                                               Global
172.16.1.
2

Global values
-----
Timeout: 3
Retransmit: 3
Deadtime: 0
Source IP: 172.16.8.1

```

# 17 Terminal Access Controller Access-Control System Plus (TACACS+) Commands

## 17.1 tacacs-server host

Use the **tacacs-server host** Global Configuration mode command to specify a TACACS+ host. Use the **no** form of this command to delete the specified TACACS+ host.

### Syntax

```
tacacs-server host {ip-address| hostname} [single-connection] [port port-number]  
[timeout timeout] [key key-string] [source {source}] [priority priority]
```

```
no tacacs-server host {ip-address| hostname}
```

### Parameters

- **host** *ip-address*—Specifies the TACACS+ server host IP address.
- **host** *hostname*—Specifies the TACACS+ server host name. (Length: 1-158 characters. Maximum label length: 63 characters)
- **single-connection**—Specifies that a single open connection is maintained between the device and the daemon, instead of the device opening and closing a TCP connection to the daemon each time it communicates.
- **port** *port-number*—Specifies the server port number. If the port number is 0, the host is not used for authentication. (Range: 0-65535)
- **timeout** *timeout*—Specifies the timeout value in seconds. (Range: 1-30)
- **key** *key-string*—Specifies the authentication and encryption key for all TACACS+ communications between the device and the TACACS+ server. This key must match the encryption used on the TACACS+ daemon. To specify an empty string, enter "". (Length: 0-128 characters)
- **source** *source*—Specifies the source IP to use for the communication. 0.0.0.0 indicates a request to use the outgoing IP interface IP address.
- **priority** *priority*—Specifies the order in which the TACACS+ servers are used, where 0 is the highest priority. (Range: 0-65535)

### Default Configuration

No TACACS+ host is specified.

The default **port-number** is 49.

If **timeout** is not specified, the global value is used.

If **key-string** is not specified, the global value is used.

If **source** is not specified, the global value is used.

### Command Mode

Global Configuration mode

### User Guidelines

Multiple **tacacs-server host** commands can be used to specify multiple hosts.

If no host-specific timeout, key, or source values are specified, the global values apply to each host.

### Example

The following example specifies a TACACS+ host.

---

```
Console(config)# tacacs-server host 172.16.1.1
```

---

## 17.2 tacacs-server key

Use the **tacacs-server key** Global Configuration mode command to set the authentication encryption key used for all TACACS+ communications between the device and the TACACS+ daemon. Use the **no** form of this command to disable the key.

### Syntax

**tacacs-server key** *key-string*

**no tacacs-server key**

### Parameters

**key-string**—Specifies the authentication and encryption key for all TACACS+ communications between the device and the TACACS+ server. This key must match the encryption used on the TACACS+ daemon. (Length: 0–128 characters)

### Default Configuration

The default key is an empty string.

### Command Mode

Global Configuration mode

### Example

The following example sets Enterprise as the authentication encryption key for all TACACS+ servers.

---

```
Console(config)# tacacs-server key enterprise
```

---

## 17.3 tacacs-server timeout

Use the **tacacs-server timeout** Global Configuration mode command to set the interval during which the device waits for a TACACS+ server to reply. Use the **no** form of this command to restore the default configuration.

### Syntax

**tacacs-server timeout** *timeout*

**no tacacs-server timeout**

### Parameters

**timeout**—Specifies the timeout value in seconds. (Range: 1-30)

### Default Configuration

The default timeout value is 5 seconds.

### Command Mode

Global Configuration mode

### Example

The following example sets the timeout value to 30 for all TACACS+ servers.

---

```
Console(config)# tacacs-server timeout 30
```

---



---

## 17.4 tacacs-server source-ip

Use the **tacacs-server source-ip** Global Configuration mode command to configure the source IP address to be used for communication with TACACS+ servers. Use the no form of this command to restore the default configuration.

### Syntax

**tacacs-server source-ip** *{source}*

**no tacacs-server source-ip** *{source}*

### Parameters

**source**—Specifies the source IP address. (Range: Valid IP address)

### Default Configuration

The default source IP address is the outgoing IP interface address.

### Command Mode

Global Configuration mode

### User Guidelines

If the configured IP source address has no available IP interface, an error message is issued when attempting to communicate with the IP address.

### Example

The following example specifies the source IP address for all TACACS+ servers.

---

```
Console(config)# tacacs-server source-ip 172.16.8.1
```

---

## 17.5 show tacacs

Use the **show tacacs** Privileged EXEC mode command to display configuration and statistical information for a TACACS+ server.

### Syntax

**show tacacs** [*ip-address*]

**Parameters**

**ip-address**—Specifies the TACACS+ server name or IP address.

**Default Configuration**

If **ip-address** is not specified, information for all TACACS+ servers is displayed.

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays configuration and statistical information for all TACACS+ servers.

---

```

Console# show tacacs

IP address      Status      Port      Single      Time      Source      Priority
-----      -
172.16.1.1     Connected   49        No          Global    Global      1

Global values
-----
Time Out: 3
Source IP: 172.16.8.1

```

---

# 18 Syslog Commands

---

## 18.1 logging on

Use the **logging on** Global Configuration mode command to control error message logging. This command sends debug or error messages asynchronously to designated locations. Use the **no** form of this command to disable the logging.

### Syntax

**logging on**

**no logging on**

### Parameters

N/A

### Default Configuration

Message logging is enabled.

### Command Mode

Global Configuration mode

### User Guidelines

The logging process controls the logging messages distribution at various destinations, such as the logging buffer, logging file or SYSLOG server. Logging on and off at these destinations can be individually configured using the [logging buffered](#), [logging file](#), and [logging on](#) Global Configuration mode commands. However, if the [logging on](#) command is disabled, no messages are sent to these destinations. Only the console receives messages.

### Example

The following example enables logging error messages.

---

```
Console(config)# logging on
```

## 18.2 logging host

Use the **logging host** Global Configuration command to log messages to the specified SYSLOG server. Use the **no** form of this command to delete the SYSLOG server with the specified address from the list of SYSLOG servers.

### Syntax

**logging host** *{ipv4-address / ipv6-address / hostname}* [**port** *port*] [**severity** *level*] [**facility** *facility*] [**description** *text*]

**no logging host** *{ipv4-address / ipv6-address / hostname}*

### Parameters

- **ipv4-address**—IPv4 address of the host to be used as a SYSLOG server.
- **ipv6-address**—Pv6 address of the host to be used as a SYSLOG server. When the IPv6 address is a Link Local address (IPv6Z address), the outgoing interface name must be specified. Refer to the User Guidelines for the interface name syntax.
- **hostname**—Hostname of the host to be used as a SYSLOG server. Only translation to IPv4 addresses is supported. (Range: 1–158 characters. Maximum label size: 63)
- **port port**—Port number for SYSLOG messages. If unspecified, the port number defaults to 514. (Range: 1–65535)
- **severity level**—Limits the logging of messages to the SYSLOG servers to a specified level: emergencies, alerts, critical, errors, warnings, notifications, informational, debugging.
- **facility facility**—The facility that is indicated in the message. It can be one of the following values: local0, local1, local2, local3, local4, local5, local 6, local7. If unspecified, the port number defaults to local7.
- **description text**—Description of the SYSLOG server. (Range: Up to 64 characters)

### Default Configuration

No messages are logged to a SYSLOG server.

if unspecified, the **severity level** defaults to Informational.

## Command Mode

Global Configuration mode

## User Guidelines

You can use multiple SYSLOG servers.

The format of an IPv6Z address is: *<ipv6-link-local-address>%<interface-name>*

- **interface-name** = *vlan<integer> | ch<integer> | isatap<integer> | /<physical-port-name> | 0*
- **integer** = *<decimal-number> | <integer><decimal-number>*
- **decimal-number** = 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
- **physical-port-name** = Designated port number, for example SYSLOG16

The following combinations are possible:

- **ipv6\_address%interface\_id** - Refers to the IPv6 address on the interface specified.
- **ipv6\_address%0** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- **ipv6\_address** - Refers to the IPv6 address on the single interface on which an IPv6 address is defined.

## Examples

```
console(config)# logging host 1.1.1.121
console(config)# logging host 3000::100/SYSLOG1
```

---

## 18.3 logging console

Use the **logging console** Global Configuration mode command to limit messages logged to the console to messages to a specific severity level. Use the **no** form of this command to restore the default.

### Syntax

**logging console** *level*

**no logging console**

### Parameters

**level**—Specifies the severity level of logged messages displayed on the console. The possible values are: emergencies, alerts, critical, errors, warnings, notifications, informational and debugging.

### Default Configuration

Informational.

### Command Mode

Global Configuration mode

### Example

The following example limits logging messages displayed on the console to messages with severity level **errors**.

---

```
Console(config)# logging console errors
```

---

## 18.4 logging buffered

Use the **logging buffered** Global Configuration mode command to limit the SYSLOG message display to messages with a specific severity level, and to define the buffer size (number of messages that can be stored). Use the **no** form of this command to cancel displaying the SYSLOG messages, and to return the buffer size to default.

### Syntax

**logging buffered** [*buffer-size*] [*severity-level* / *severity-level-name*]

**no logging buffered**

### Parameters

- **buffer-size**—Specifies the maximum number of messages stored in the history table. (Range: 20–400)
- **severity-level**—Specifies the severity level of messages logged in the buffer. The possible values are: 1-7.
- **severity-level-name**—Specifies the severity level of messages logged in the buffer. The possible values are: emergencies, alerts, critical, errors, warnings, notifications, informational and debugging.

### Default Configuration

The default severity level is informational.

The default buffer size is 200.

### Command Mode

Global Configuration mode

### User Guidelines

All the SYSLOG messages are logged to the internal buffer. This command limits the messages displayed to the user.

### Example

The following example shows two ways of limiting the SYSLOG message display from an internal buffer to messages with severity level **debugging**. In the second example, the buffer size is set to 100.

---

```
Console(config)# logging buffered debugging
```

```
Console(config)# logging buffered 100 7
```

---

## 18.5 clear logging

Use the **clear logging** Privileged EXEC mode command to clear messages from the internal logging buffer.

### Syntax

**clear logging**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### Example

The following example clears messages from the internal logging buffer.

---

```
Console# clear logging
Clear logging buffer [confirm]
```

---

## 18.6 logging file

Use the **logging file** Global Configuration mode command to limit SYSLOG messages sent to the logging file to messages with a specific severity level. Use the **no** form of this command to cancel sending messages to the file.

### Syntax

**logging file** *level*

**no logging file**

### Parameters

**level**—Specifies the severity level of SYSLOG messages sent to the logging file. The possible values are: emergencies, alerts, critical, errors, warnings, notifications, informational and debugging.

### Default Configuration

The default severity level is **errors**.

### Command Mode

Global Configuration mode

### Example

The following example limits SYSLOG messages sent to the logging file to messages with severity level **alerts**.

---

```
Console(config)# logging file alerts
```

---



---

## 18.7 clear logging file

Use the **clear logging file** Privileged EXEC mode command to clear messages from the logging file.

### Syntax

**clear logging file**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### Example

The following example clears messages from the logging file.

---

```
Console# clear logging file
Clear Logging File [y/n]
```

---

## 18.8 file-system logging

Use the **file-system logging** Global Configuration mode command to enable logging file system events. Use the **no** form of this command to disable logging file system events.

### Syntax

**file-system logging** *{copy | delete-rename}*

**no file-system logging** *{copy | delete-rename}*

### Parameters

- **copy**—Specifies logging messages related to file copy operations.

- **delete-rename**—Specifies logging messages related to file deletion and renaming operations.

### Default Configuration

Enabled.

### Command Mode

Global Configuration mode

### Example

The following example enables logging messages related to file copy operations.

---

```
Console(config)# file-system logging copy
```

---

## 18.9 logging aggregation on

Use the **logging aggregation on** Global Configuration mode command to control aggregation of SYSLOG messages. If aggregation is enabled, logging messages are displayed every time interval (according to the aging time specified by [logging aggregation aging-time](#)). Use the **no** form of this command to disable aggregation of SYSLOG messages.

### Syntax

**logging aggregation on**

**no logging aggregation on**

### Parameters

N/A

### Default Configuration

Enabled.

### Command Mode

Global Configuration mode

### Example

To turn off aggregation of SYSLOG messages:

---

```
console(config)# no logging aggregation on
```

---

## 18.10 logging aggregation aging-time

Use the **logging aggregation aging-time** Global Configuration mode command to configure the aging time of the aggregated SYSLOG messages. The SYSLOG messages are aggregated during the time interval set by the aging-time parameter. Use the **no** form of this command to return to the default.

### Syntax

**logging aggregation aging-time** *sec*

**no logging aggregation aging-time**

### Parameters

**aging-time** *sec*—Aging time in seconds (Range: 15–3600)

### Default Configuration

300 seconds.

### Command Mode

Global Configuration mode

### Example

---

```
console(config)# logging aggregation aging-time 300
```

---

## 18.11 show logging

Use the **show logging** Privileged EXEC mode command to display the logging status and SYSLOG messages stored in the internal buffer.

### Syntax

**show logging**

## Parameters

N/A

## Default Configuration

N/A

## Command Mode

Privileged EXEC mode

## Example

The following example displays the logging status and the SYSLOG messages stored in the internal buffer.

---

```
console# show logging
Logging is enabled.
Console Logging: Level info. Console Messages: 0 Dropped.
Buffer Logging: Level info. Buffer Messages: 61 Logged, 61 Displayed, 200 Max.
File Logging: Level error. File Messages: 898 Logged, 64 Dropped.
4 messages were not logged

Application filtering control
Application          Event                Status
-----
AAA                  Login                 Enabled
File system          Copy                  Enabled
File system          Delete-Rename        Enabled
Management ACL       Deny                  Enabled

Aggregation: Disabled.
Aggregation aging time: 300 Sec

01-Jan-2010 05:29:46 :%INIT-I-Startup: Warm Startup
```

```
01-Jan-2010 05:29:02 :%LINK-I-Up:  Vlan 1
01-Jan-2010 05:29:02 :%LINK-I-Up:  SYSLOG6
01-Jan-2010 05:29:02 :%LINK-I-Up:  SYSLOG7
01-Jan-2010 05:29:00 :%LINK-W-Down:  SYSLOG8
```

---

## 18.12 show logging file

Use the **show logging file** Privileged EXEC mode command to display the logging status and the SYSLOG messages stored in the logging file.

### Syntax

**show logging file**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### Example

The following example displays the logging status and the SYSLOG messages stored in the logging file.

---

```
console# show logging file
Logging is enabled.

Console Logging: Level info. Console Messages: 0 Dropped.

Buffer Logging: Level info. Buffer Messages: 61 Logged, 61 Displayed, 200 Max.

File Logging: Level error. File Messages: 898 Logged, 64 Dropped.

4 messages were not logged

Application filtering control
```

Application	Event	Status
-----	-----	-----
AAA	Login	Enabled
File system	Copy	Enabled
File system	Delete-Rename	Enabled
Management ACL	Deny	Enabled

Aggregation: Disabled.

Aggregation aging time: 300 Sec

01-Jan-2010 05:57:00 :%SSHD-E-ERROR: SSH error: key\_read: type mismatch: encoding error

01-Jan-2010 05:56:36 :%SSHD-E-ERROR: SSH error: key\_read: type mismatch: encoding error

01-Jan-2010 05:55:37 :%SSHD-E-ERROR: SSH error: key\_read: type mismatch: encoding error

01-Jan-2010 05:55:03 :%SSHD-E-ERROR: SSH error: key\_read: key\_from\_blob bgEgGnt9z6NHgZwKI5xKqF7cBtdl1xmFgSEWuDhho5UedydAjVkKS5XR2... failed

01-Jan-2010 05:55:03 :%SSHD-E-ERROR: SSH error: key\_from\_blob: invalid key type.

01-Jan-2010 05:56:34 :%SSHD-E-ERROR: SSH error: bad sigbloblen 58 != SIGBLOB\_LEN  
console#

### 18.13 show syslog-servers

Use the **show syslog-servers** Privileged EXEC mode command to display the SYSLOG server settings.

**Syntax****show syslog-servers****Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

Privileged EXEC mode

**Example**

The following example provides information about the SYSLOG servers.

---

```
console# show syslog-servers
```

```
Device Configuration
```

```
IP address      Port    Facility Severity  Description
```

```
-----
```

```
1.1.1.121      514    local7   info
```

```
3000::100     514    local7   info
```

# 19 Remote Network Monitoring (RMON) Commands

## 19.1 show rmon statistics

Use the **show rmon statistics** EXEC mode command to display RMON Ethernet statistics.

### Syntax

**show rmon statistics** *{interface-id}*

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

### Command Mode

EXEC mode

### Example

The following example displays RMON Ethernet statistics for gigabitethernet port gi 1.

```
console# show rmon statistics gi1
Port gi1
Dropped: 0
Octets: 0                               Packets: 0
Broadcast: 0                             Multicast: 0
CRC Align Errors: 0                       Collisions: 0
Undersize Pkts: 0                         Oversize Pkts: 0
Fragments: 0                              Jabbers: 0
64 Octets: 0                              65 to 127 Octets: 1
128 to 255 Octets: 1                       256 to 511 Octets: 1
512 to 1023 Octets: 0                      1024 to max Octets: 0
```



The following table describes the significant fields displayed.

Field	Description
<b>Dropped</b>	The total number of events in which packets were dropped by the probe due to lack of resources. Note that this number is not necessarily the number of packets dropped. It is the number of times this condition was detected.
<b>Octets</b>	The total number of octets of data (including those in bad packets) received on the network (excluding framing bits but including FCS octets).
<b>Packets</b>	The total number of packets (including bad packets, broadcast packets, and multicast packets) received.
<b>Broadcast</b>	The total number of good packets received and directed to the broadcast address. This does not include multicast packets.
<b>Multicast</b>	The total number of good packets received and directed to a multicast address. This number does not include packets directed to the broadcast address.
<b>CRC Align Errors</b>	The total number of packets received with a length (excluding framing bits, but including FCS octets) of between 64 and 1518 octets, inclusive, but with either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error).
<b>Collisions</b>	The best estimate of the total number of collisions on this Ethernet segment.
<b>Undersize Pkts</b>	The total number of packets received, less than 64 octets long (excluding framing bits, but including FCS octets) and otherwise well formed.
<b>Oversize Pkts</b>	The total number of packets received, longer than 1518 octets (excluding framing bits, but including FCS octets) and otherwise well formed.

Field	Description
<b>Fragments</b>	The total number of packets received, less than 64 octets in length (excluding framing bits but including FCS octets) and either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error).
<b>Jabbers</b>	The total number of packets received, longer than 1518 octets (excluding framing bits, but including FCS octets), and either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error).
<b>64 Octets</b>	The total number of packets (including bad packets) received that are 64 octets in length (excluding framing bits but including FCS octets).
<b>65 to 127 Octets</b>	The total number of packets (including bad packets) received that are between 65 and 127 octets in length inclusive (excluding framing bits but including FCS octets).
<b>128 to 255 Octets</b>	The total number of packets (including bad packets) received that are between 128 and 255 octets in length inclusive (excluding framing bits but including FCS octets).
<b>256 to 511 Octets</b>	The total number of packets (including bad packets) received that are between 256 and 511 octets in length inclusive (excluding framing bits but including FCS octets).
<b>512 to 1023 Octets</b>	The total number of packets (including bad packets) received that were between 512 and 1023 octets in length inclusive (excluding framing bits but including FCS octets).
<b>1024 to max</b>	The total number of packets (including bad packets) received that were between 1024 octets and the maximum frame size in length inclusive (excluding framing bits but including FCS octets).

## 19.2 rmon collection stats

Use the **rmon collection stats** Interface Configuration mode command to enable Remote Monitoring (RMON) MIB history group of statistics on an interface. Use the **no** form of this command to remove a specified RMON history group of statistics.

### Syntax

**rmon collection stats** index [*owner* *ownername*] [*buckets* *bucket-number*] [*interval* *seconds*]

**no rmon collection stats** *index*

### Parameters

- **index**—The requested group of statistics index.(Range: 1–65535)
- **owner** *ownername*—Records the name of the owner of the RMON group of statistics. If unspecified, the name is an empty string. (Range: Valid string)
- **buckets** *bucket-number*—A value associated with the number of buckets specified for the RMON collection history group of statistics. If unspecified, defaults to 50.(Range: 1–50)
- **interval** *seconds*—The number of seconds in each polling cycle. If unspecified, defaults to 1800 (Range: 1–3600).

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode. Cannot be configured for a range of interfaces (range context).

---

## 19.3 show rmon collection stats

Use the **show rmon collection stats** EXEC mode command to display the requested RMON history group statistics.

### Syntax

**show rmon collection stats** [*interface-id*]

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

### Command Mode

EXEC mode

**Example**

The following example displays all RMON history group statistics.

```

Console# show rmon collection stats
Index   Interface  Interval  Requested  Granted  Owner
                   Samples   Samples
-----  -
1       gil       30        50         50      CLI
2       gil      1800     50         50     Manager

```

The following table describes the significant fields shown in the display.

Field	Description
<b>Index</b>	An index that uniquely identifies the entry.
<b>Interface</b>	The sampled Ethernet interface.
<b>Interval</b>	The interval in seconds between samples.
<b>Requested Samples</b>	The requested number of samples to be saved.
<b>Granted Samples</b>	The granted number of samples to be saved.
<b>Owner</b>	The entity that configured this entry.

**19.4 show rmon history**

Use the **show rmon history** EXEC mode command to display RMON Ethernet history statistics.

**Syntax**

```
show rmon history index {throughput | errors | other} [period seconds]
```

**Parameters**

- **index**—Specifies the set of samples to display. (Range: 1–65535)
- **throughput**—Displays throughput counters.
- **errors**—Displays error counters.
- **other**—Displays drop and collision counters.

- **period *seconds***—Specifies the period of time in seconds to display. (Range: 1–2147483647)

### Command Mode

EXEC mode

### Example

The following examples display RMON Ethernet history statistics for index 1

---

```

Console# show rmon history 1 throughput
Sample Set: 1                Owner: CLI
Interface: gil              Interval: 1800
Requested samples: 50       Granted samples: 50

Maximum table size: 500
Time                Octets    Packets  Broadcas  Multicast  Util
-----            -
Jan 18 2005        30359596  357568  -----  7287       19%
21:57:00           2         275686  3289     5878       20%
Jan 18 2005        28769630  2789
21:57:30           4

Console# show rmon history 1 errors
Sample Set: 1                Owner: Me
Interface:gil              Interval: 1800
Requested samples: 50       Granted samples: 50

Maximum table size: 500 (800 after reset)

```

```

                CRC      Under
Time           Align   size   Oversize  Fragments  Jabbers
-----
Jan 18 2005   1         1       0         49         0
21:57:00     1         1       0         27         0

Jan 18 2005
21:57:30
Console# show rmon history 1 other
Sample Set: 1                               Owner: Me
Interface: gil                               Interval: 1800
Requested samples: 50                       Granted samples: 50
Maximum table size: 500
Time                                           Dropped  Collisions
-----
Jan 18 2005 21:57:00                         3         0
Jan 18 2005 21:57:30                         3         0

```

The following table describes significant fields shown in the display:

Field	Description
<b>Time</b>	Date and Time the entry is recorded.
<b>Octets</b>	The total number of octets of data (including those in bad packets and excluding framing bits but including FCS octets) received on the network.
<b>Packets</b>	The number of packets (including bad packets) received during this sampling interval.
<b>Broadcast</b>	The number of good packets received during this sampling interval that were directed to the broadcast address.
<b>Multicast</b>	The number of good packets received during this sampling interval that were directed to a multicast address. This number does not include packets addressed to the broadcast address.
<b>Utilization</b>	The best estimate of the mean physical layer network utilization on this interface during this sampling interval, in hundredths of a percent.

Field	Description
<b>CRC Align</b>	The number of packets received during this sampling interval that had a length (excluding framing bits but including FCS octets) between 64 and 1518 octets, inclusive, but had either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error).
<b>Undersize</b>	The number of packets received during this sampling interval that were less than 64 octets long (excluding framing bits but including FCS octets) and were otherwise well formed.
<b>Oversize</b>	The number of packets received during this sampling interval that were longer than 1518 octets (excluding framing bits but including FCS octets) but were otherwise well formed.
<b>Fragments</b>	The total number of packets received during this sampling interval that were less than 64 octets in length (excluding framing bits but including FCS octets) and had either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error), or a bad FCS with a non-integral number of octets (Alignment Error). It is normal for etherHistoryFragments to increment because it counts both runts (which are normal occurrences due to collisions) and noise hits.
<b>Jabbers</b>	The number of packets received during this sampling interval that were longer than 1518 octets (excluding framing bits but including FCS octets), and had either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error).
<b>Dropped</b>	The total number of events in which packets were dropped by the probe due to lack of resources during this sampling interval. This number is not necessarily the number of packets dropped, it is the number of times this condition has been detected.
<b>Collisions</b>	The best estimate of the total number of collisions on this Ethernet segment during this sampling interval.

## 19.5 rmon alarm

Use the **rmon alarm** Global Configuration mode command to configure alarm conditions. Use the **no** form of this command to remove an alarm.

## Syntax

```
rmon alarm index mib-object-id interval rthreshold fthreshold revent fevent [type  
{absolute | delta}]  
[startup {rising | rising-falling | falling}] [owner name]  
no rmon alarm index
```

## Parameters

- **index**—Specifies the alarm index. (Range: 1–65535)
- **mib-object-id**—Specifies the object identifier of the variable to be sampled. (Valid OID)
- **interval**—Specifies the interval in seconds during which the data is sampled and compared with rising and falling thresholds. (Range: 1–4294967295)
- **rthreshold**—Specifies the rising threshold value. (Range: 0–4294967295)
- **fthreshold**—Specifies the falling threshold value. (Range: 0–4294967295)
- **revent**—Specifies the index of the event triggered when a rising threshold is crossed. (Range: 0–65535)
- **fevent**—Specifies the index of the event triggered when a falling threshold is crossed. (Range: 0–65535)
- **type {absolute | delta}**—Specifies the method used for sampling the selected variable and calculating the value to be compared against the thresholds. The possible values are:
  - **absolute**—Specifies that the selected variable value is compared directly with the thresholds at the end of the sampling interval.
  - **delta**—Specifies that the selected variable value of the last sample is subtracted from the current value, and the difference is compared with the thresholds.
- **startup {rising | rising-falling | falling}**—Specifies the alarm that may be sent when this entry becomes valid. The possible values are:
  - **rising**—Specifies that if the first sample (after this entry becomes valid) is greater than or equal to **rthreshold**, a single rising alarm is generated.
  - **rising-falling**—Specifies that if the first sample (after this entry becomes valid) is greater than or equal to **rthreshold**, a single rising alarm is



generated. If the first sample (after this entry becomes valid) is less than or equal to **fthreshold**, a single falling alarm is generated.

- **falling** —Specifies that if the first sample (after this entry becomes valid) is less than or equal to **fthreshold**, a single falling alarm is generated.
- **owner name**—Specifies the name of the person who configured this alarm. (Valid string)

### Default Configuration

The default method type is **absolute**.

The default startup direction is **rising-falling**.

If the owner **name** is not specified, it defaults to an empty string.

### Command Mode

Global Configuration mode

### Example

The following example configures an alarm with index 1000, MIB object ID D-Link, sampling interval 360000 seconds (100 hours), rising threshold value 1000000, falling threshold value 1000000, rising threshold event index 10, falling threshold event index 10, absolute method type and rising-falling alarm.

```
console(config)# rmon alarm 1000 1.3.6.1.2.1.2.2.1.10.1 360000 1000000 1000000 10
20
```

## 19.6 show rmon alarm-table

Use the **show rmon alarm-table** EXEC mode command to display a summary of the alarms table.

### Syntax

**show rmon alarm-table**

### Command Mode

EXEC mode

**Example**

The following example displays the alarms table.

---

```

Console# show rmon alarm-table

Index      OID                               Owner
-----
1          1.3.6.1.2.1.2.2.1.10.1          CLI
2          1.3.6.1.2.1.2.2.1.10.1          Manager
3          1.3.6.1.2.1.2.2.1.10.9          CLI

```

The following table describes the significant fields shown in the display:

Field	Description
<b>Index</b>	An index that uniquely identifies the entry.
<b>OID</b>	Monitored variable OID.
<b>Owner</b>	The entity that configured this entry.

---

## 19.7 show rmon alarm

Use the **show rmon alarm** EXEC mode command to display alarm configuration.

**Syntax**

**show rmon alarm** *number*

**Parameters**

**alarm** *number*—Specifies the alarm index. (Range: 1–65535)

**Command Mode**

EXEC mode

**Example**

The following example displays RMON 1 alarms.

---

```

Console# show rmon alarm 1

```

```

Alarm 1
-----
OID: 1.3.6.1.2.1.2.2.1.10.1

Last sample Value: 878128

Interval: 30

Sample Type: delta

Startup Alarm: rising

Rising Threshold: 8700000

Falling Threshold: 78

Rising Event: 1

Falling Event: 1

Owner: CLI

```

The following table describes the significant fields shown in the display:

Field	Description
<b>Alarm</b>	Alarm index.
<b>OID</b>	Monitored variable OID.
<b>Last Sample Value</b>	The value of the statistic during the last sampling period. For example, if the sample type is <b>delta</b> , this value is the difference between the samples at the beginning and end of the period. If the sample type is <b>absolute</b> , this value is the sampled value at the end of the period.
<b>Interval</b>	The interval in seconds over which the data is sampled and compared with the rising and falling thresholds.
<b>Sample Type</b>	The method of sampling the variable and calculating the value compared against the thresholds. If the value is <b>absolute</b> , the variable value is compared directly with the thresholds at the end of the sampling interval. If the value is <b>delta</b> , the variable value at the last sample is subtracted from the current value, and the difference is compared with the thresholds.

Field	Description
<b>Startup Alarm</b>	The alarm that may be sent when this entry is first set. If the first sample is greater than or equal to the rising threshold, and startup alarm is equal to rising or rising-falling, then a single rising alarm is generated. If the first sample is less than or equal to the falling threshold, and startup alarm is equal falling or rising-falling, then a single falling alarm is generated.
<b>Rising Threshold</b>	The sampled statistic rising threshold. When the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval is less than this threshold, a single event is generated.
<b>Falling Threshold</b>	The sampled statistic falling threshold. When the current sampled value is less than or equal to this threshold, and the value at the last sampling interval is greater than this threshold, a single event is generated.
<b>Rising Event</b>	The event index used when a rising threshold is crossed.
<b>Falling Event</b>	The event index used when a falling threshold is crossed.
<b>Owner</b>	The entity that configured this entry.

## 19.8 rmon event

Use the **rmon event** Global Configuration mode command to configure an event. Use the **no** form of this command to remove an event.

### Syntax

```
rmon event index {none | log | trap | log-trap} [community text] [description text] [owner name]
```

```
no rmon event index
```

### Parameters

- **index**—Specifies the event index. (Range: 1–65535)
- **none**— Specifies that no notification is generated by the device for this event.

- **log**—Specifies that a notification entry is generated in the log table by the device for this event.
- **trap**—Specifies that an SNMP trap is sent to one or more management stations by the device for this event.
- **log-trap**—Specifies that an entry is generated in the log table and an SNMP trap is sent to one or more management stations by the device for this event.
- **community text**—Specifies the SNMP community to which an SNMP trap is sent. (Octet string; length: 0–127 characters)
- **description text**—Specifies a comment describing this event. (Length: 0–127 characters)
- **owner name**—Specifies the name of the person who configured this event. (Valid string)

### Default Configuration

If the owner name is not specified, it defaults to an empty string.

### Command Mode

Global Configuration mode

### Example

The following example configures an event identified as index 10, for which the device generates a notification in the log table.

---

```
Console(config)# rmon event 10 log
```

---

## 19.9 show rmon events

Use the **show rmon events** EXEC mode command to display the RMON event table.

### Syntax

**show rmon events**

**Command Mode**

EXEC mode

**Example**

The following example displays the RMON event table.

```

Console# show rmon events
-----
Index   Description      Type      Community  Owner      Last time sent
-----
1       Errors           Log       router     CLI        Jan 18 2006 23:58:17
2       High Broadcast    Log       Trap       Manager    Jan 18 2006 23:59:48

```

The following table describes significant fields shown in the display:

Field	Description
<b>Index</b>	A unique index that identifies this event.
<b>Description</b>	A comment describing this event.
<b>Type</b>	The type of notification that the device generates about this event. Can have the following values: <b>none</b> , <b>log</b> , <b>trap</b> , <b>log-trap</b> . In the case of log, an entry is made in the log table for each event. In the case of trap, an SNMP trap is sent to one or more management stations.
<b>Community</b>	If an SNMP trap is to be sent, it is sent to the SNMP community specified by this octet string.
<b>Owner</b>	The entity that configured this event.
<b>Last time sent</b>	The time this entry last generated an event. If this entry has not generated any events, this value is zero.

**19.10 show rmon log**

Use the **show rmon log** EXEC mode command to display the RMON log table.

**Syntax**

```
show rmon log [event]
```

**Parameters**

**event**—Specifies the event index. (Range: 0–65535)

**Command Mode**

EXEC mode

**Example**

The following example displays event 1 in the RMON log table.

---

```

Console# show rmon log 1

Maximum table size: 500 (800 after reset)
Event          Description                Time
-----
1              MIB Var.:                    Jan 18 2006 23:48:19
              1.3.6.1.2.1.2.2.1.10.
              53, Delta, Rising,
              Actual Val: 800,
              Thres.Set: 100,
              Interval (sec):1

```

---

**19.11 rmon table-size**

Use the **rmon table-size** Global Configuration mode command to configure the maximum size of RMON tables. Use the no form of this command to return to the default configuration.

**Syntax**

**rmon table-size** *{history entries | log entries}*

**no rmon table-size** *{history | log}*

**Parameters**

- **history entries**—Specifies the maximum number of history table entries. (Range: 20–270)
- **log entries**—Specifies the maximum number of log table entries. (Range: 20–100)

**Default Configuration**

The default history table size is 270 entries.

The default log table size is 200 entries.

**Command Mode**

Global Configuration mode

**User Guidelines**

The configured table size takes effect after the device is rebooted.

**Example**

The following example configures the maximum size of RMON history tables to 100 entries.

---

```
Console(config)# rmon table-size history 100
```



## 20 802.1x Commands

### 20.1 aaa authentication dot1x

Use the **aaa authentication dot1x** Global Configuration mode command to specify how ports are authenticated when 802.1x is enabled. You can select either authentication by a RADIUS server, no authentication, or both methods. Use the **no** form of this command to restore the default configuration.

#### Syntax

```
aaa authentication dot1x default method1 [method2]
```

```
no aaa authentication dot1x default
```

#### Parameters

**method1** [**method2**]  
—Specify at least one method from the following list:

Keyword	Description
<b>radius</b>	Uses the list of all RADIUS servers for authentication
<b>none</b>	Uses no authentication

#### Default Configuration

The default method is Radius.

#### Command Mode

Global Configuration mode

#### User Guidelines

You can select either authentication by a RADIUS server, no authentication (**none**), or both methods. If both RADIUS and **none** are selected, authentication begins with the RADIUS server. If no RADIUS server answers, no authentication is performed.

If you require that authentication succeeds even the RADIUS server is not found or returns an error, specify **none** as the final method in the command line.

### Example

The following example sets 802.1X authentication mode to RADIUS server authentication. If no response is received, no authentication is performed.

```
Console(config)# aaa authentication dot1x default radius none
```

---

## 20.2 dot1x system-auth-control

Use the **dot1x system-auth-control** Global Configuration mode command to enable 802.1x globally. Use the **no** form of this command to restore the default configuration.

### Syntax

**dot1x system-auth-control**

**no dot1x system-auth-control**

### Default Configuration

All the ports are in FORCE\_AUTHORIZED state.

### Command Mode

Global Configuration mode

### Example

The following example enables 802.1x globally.

```
Console(config)# dot1x system-auth-control
```

---

## 20.3 dot1x port-control

Use the **dot1x port-control** Interface Configuration (Ethernet) mode command to enable manual control of the port authorization state. Use the **no** form of this command to restore the default configuration.

### Syntax

**dot1x port-control** *{auto / force-authorized / force-unauthorized}[time-range time-range-name]*

## no dot1x port-control

### Parameters

- **auto**—Enables 802.1x authentication on the interface and causes the port to transition to the authorized or unauthorized state based on the 802.1x authentication exchange between the device and the client.
- **force-authorized**—Disables 802.1x authentication on the interface and causes the port to transition to the authorized state without any authentication exchange required. The port resends and receives normal traffic without 802.1x-based client authentication.
- **force-unauthorized**—Denies all access through this interface by forcing the port to transition to the unauthorized state and ignoring all attempts by the client to authenticate. The device cannot provide authentication services to the client through the interface.
- **time-range** time-range-name—Specifies a time range. When the Time Range is not in effect, the port state is Unauthorized. (Range: 1–32 characters)

### Default Configuration

The port is in the force-authorized state.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

It is recommended to disable spanning tree or to enable spanning-tree PortFast mode on 802.1x edge ports (ports in **auto** state that are connected to end stations), in order to proceed to the forwarding state immediately after successful authentication.

### Example

The following example enables 802.1x authentication on gi 15.

---

```
Console(config)# interface gi15
Console(config-if)# dot1x port-control auto
```

---

## 20.4 dot1x reauthentication

Use the **dot1x reauthentication** Interface Configuration mode command to enable periodic re-authentication of the client. Use the **no** form of this command to return to the default setting.

### Syntax

**dot1x reauthentication**

**no dot1x reauthentication**

### Parameters

N/A

### Default Configuration

Periodic re-authentication is disabled.

### Command Mode

Interface configuration (Ethernet)

### Example

---

```
console(config)# interface gil
console(config-if)# dot1x reauthentication
```

---

## 20.5 dot1x timeout reauth-period

Use the **dot1x timeout reauth-period** Interface Configuration mode command to set the number of seconds between re-authentication attempts. Use the **no** form of this command to return to the default setting.

### Syntax

**dot1x timeout reauth-period** *seconds*

**no dot1x timeout reauth-period**

### Parameters

**reauth-period** *seconds*—Number of seconds between re-authentication attempts. (Range: 300-4294967295)

### Default Configuration

3600

### Command Mode

Interface Configuration (Ethernet) mode

### Example

```
console(config)# interface gil
console(config-if)# dot1x timeout reauth-period 5000
```

---

## 20.6 dot1x timeout quiet-period

Use the **dot1x timeout quiet-period** Interface Configuration (Ethernet) mode command to set the time interval that the device remains in a quiet state following a failed authentication exchange (for example, the client provided an invalid password). Use the **no** form of this command to restore the default configuration.

### Syntax

**dot1x timeout quiet-period** *seconds*

**no dot1x timeout quiet-period**

### Parameters

**seconds**—Specifies the time interval in seconds that the device remains in a quiet state following a failed authentication exchange with the client. (Range: 30–65535 seconds)

### Default Configuration

The default quiet period is 60 seconds.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

During the quiet period, the device does not accept or initiate authentication requests.

The default value of this command should only be changed to adjust to unusual circumstances, such as unreliable links or specific behavioral problems with certain clients and authentication servers.

To provide faster response time to the user, a smaller number than the default value should be entered.

### Example

The following example sets the time interval that the device remains in the quiet state following a failed authentication exchange to 3600 seconds.

```
Console(config)# interface gi15
Console(config-if)# dot1x timeout quiet-period 3600
```

## 20.7 dot1x timeout tx-period

Use the **dot1x timeout tx-period** Interface Configuration (Ethernet) mode command to set the time interval during which the device waits for a response to an Extensible Authentication Protocol (EAP) request/identity frame from the client before resending the request. Use the **no** form of this command to restore the default configuration.

### Syntax

**dot1x timeout tx-period** *seconds*

**no dot1x timeout tx-period**

### Parameters

**seconds**—Specifies the time interval in seconds during which the device waits for a response to an EAP-request/identity frame from the client before resending the request. (Range: 1–65535 seconds)

### Default Configuration

The default timeout period is 30 seconds.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

The default value of this command should be changed only to adjust to unusual circumstances, such as unreliable links or specific behavioral problems with certain clients and authentication servers.

### Example

The following command sets the time interval during which the device waits for a response to an EAP request/identity frame to 3600 seconds.

```
Console(config)# interface gi15
Console(config-if)# dot1x timeout tx-period 3600
```

---

## 20.8 dot1x max-req

Use the **dot1x max-req** Interface Configuration mode command to set the maximum number of times that the device sends an Extensible Authentication Protocol (EAP) request/identity frame (assuming that no response is received) to the client before restarting the authentication process. Use the **no** form of this command to restore the default configuration.

### Syntax

**dot1x max-req** *count*

**no dot1x max-req**

### Parameters

**max-req** *count*—Specifies the maximum number of times that the device sends an EAP request/identity frame before restarting the authentication process. (Range: 1–10)

### Default Configuration

The default maximum number of attempts is 2.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

The default value of this command should be changed only to adjust to unusual circumstances, such as unreliable links or specific behavioral problems with certain clients and authentication servers.

### Example

The following example sets the maximum number of times that the device sends an EAP request/identity frame to 6

```
Console(config)# interface gi15
Console(config-if)# dot1x max-req 6
```

---

## 20.9 dot1x timeout supp-timeout

Use the **dot1x timeout supp-timeout** Interface Configuration (Ethernet) mode command to set the time interval during which the device waits for a response to an Extensible Authentication Protocol (EAP) request frame from the client before resending the request. Use the **no** form of this command to restore the default configuration.

### Syntax

**dot1x timeout supp-timeout** *seconds*

**no dot1x timeout supp-timeout**

### Parameters

**supp-timeout** *seconds*—Specifies the time interval in seconds during which the device waits for a response to an EAP request frame from the client before resending the request. (Range: 1–65535 seconds)

### Default Configuration

The default timeout period is 30 seconds.

### Command Mode

Interface Configuration (Ethernet) mode



### User Guidelines

The default value of this command should be changed only to adjust to unusual circumstances, such as unreliable links or specific behavioral problems with certain clients and authentication servers.

### Example

The following example sets the time interval during which the device waits for a response to an EAP request frame from the client before resending the request to 3600 seconds.

```
Console(config)# interface gi15
Console(config-if)# dot1x timeout supp-timeout 3600
```

---

## 20.10 dot1x timeout server-timeout

Use the **dot1x timeout server-timeout** Interface Configuration (Ethernet) mode command to set the time interval during which the device waits for a response from the authentication server. Use the **no** form of this command to restore the default configuration.

### Syntax

**dot1x timeout server-timeout** *seconds*

**no dot1x timeout server-timeout**

### Parameters

**server-timeout** *seconds*—Specifies the time interval in seconds during which the device waits for a response from the authentication server. (Range: 1–65535 seconds)

### Default Configuration

The default timeout period is 30 seconds.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

The actual timeout period can be determined by comparing the value specified by the **dot1x timeout server-timeout** command to the result of multiplying the number of retries specified by the **radius-server retransmit** command by the timeout period specified by the **radius-server timeout** command, and selecting the lower of the two values.

### Example

The following example sets the time interval between retransmission of packets to the authentication server to 3600 seconds.

```
Console(config)# interface gil5
Console(config-if)# dot1x timeout server-timeout 3600
```

## 20.11 show dot1x

Use the **show dot1x** Privileged EXEC mode command to display the 802.1x device or specified interface status.

### Syntax

```
show dot1x [interface interface-id]
```

### Parameters

**interface-id**—Specify an interface ID. The interface ID must be an Ethernet port.

### Command Mode

Privileged EXEC mode

**Example**

The following examples display the status of 802.1x-enabled Ethernet ports.

---

```
Console# show dot1x
```

```
802.1x is enabled
```

Port	Admin Mode	Oper Mode	Reauth Control	Reauth Period	Username
----	-----	-----	-----	-----	-----
gi1	Auto	Authorized	Ena	3600	Bob
gi2	Auto	Authorized	Ena	3600	John
gi3	Auto	Unauthorized	Ena	3600	Clark
gi4	Force-auth	Authorized	Dis	3600	n/a
gi5	Force-auth	Unauthorized	Dis	3600	n/a

```
* Port is down or not present.
```

```
Console# show dot1x interface gi3
```

```
802.1x is enabled.
```

Port	Admin Mode	Oper Mode	Reauth Control	Reauth Period	Username
----	-----	-----	-----	-----	-----
gi3	Auto	Unauthorized	Ena	3600	Clark

```

Time-range:          work-hours (Inactive now)
Quiet period:        60 Seconds
Tx period:           30 Seconds
Max req:             2
Supplicant timeout:  30 Seconds
Server timeout:     30 Seconds
Session Time (HH:MM:SS): 08:19:17
MAC Address:         00:08:78:32:98:78
Authentication Method: Remote
Termination Cause:  Supplicant logoff

Authenticator State Machine
State:               HELD

Backend State Machine
State:               IDLE

Authentication success: 9
Authentication fails:  1

```

The following table describes the significant fields shown in the display.

Field	Description
<b>Port</b>	The port number.
<b>Admin mode</b>	The port administration (configured) mode. Possible values: Force-auth, Force-unauth, Auto.
<b>Oper mode</b>	The port operational (actual) mode. Possible values: Authorized, Unauthorized or Down.
<b>Reauth Control</b>	Reauthentication control.
<b>Reauth Period</b>	Reauthentication period.

Field	Description
<b>Username</b>	The username representing the supplicant identity. This field shows the username if the port control is auto. If the port is Authorized, it displays the username of the current user. If the port is Unauthorized, it displays the last user authenticated successfully.
<b>Quiet period</b>	The number of seconds that the device remains in the quiet state following a failed authentication exchange (for example, the client provided an invalid password).
<b>Tx period</b>	The number of seconds that the device waits for a response to an Extensible Authentication Protocol (EAP) request/identity frame from the client before resending the request.
<b>Max req</b>	The maximum number of times that the device sends an EAP request frame (assuming that no response is received) to the client before restarting the authentication process.
<b>Supplicant timeout</b>	The number of seconds that the device waits for a response to an EAP-request frame from the client before resending the request.
<b>Server timeout</b>	The number of seconds that the device waits for a response from the authentication server before resending the request.
<b>Session Time</b>	The amount of time (HH:MM:SS) that the user is logged in.
<b>MAC address</b>	The supplicant MAC address.
<b>Authentication Method</b>	The authentication method used to establish the session.
<b>Termination Cause</b>	The reason for the session termination.
<b>State</b>	The current value of the Authenticator PAE state machine and of the Backend state machine.
<b>Authentication success</b>	The number of times the state machine received a Success message from the Authentication Server.
<b>Authentication fails</b>	The number of times the state machine received a Failure message from the Authentication Server.

---

## 20.12 show dot1x users

Use the **show dot1x users** Privileged EXEC mode command to display active 802.1x authenticated users for the device.

### Syntax

**show dot1x users** [*username username*]

### Parameters

**username**—Specifies the supplicant username (Length: 1–160 characters)

### Command Mode

Privileged EXEC mode

**Example**

The following example displays 802.1x users.

```
Switch# show dot1x users
```

Port	Username	Session Time	Auth Method	MAC Address	VLAN	Filter
gi1	Bob	1d 03:08:58	Remote	0008.3b79.8787	3	
gi2	John	08:19:17	None	0008.3b89.3127	2	OK

```
Switch# show dot1x users
```

Port	Username	Session Time	Auth Method	MAC Address	VLAN	Filter
gi1	Bob	1d 09:07:38	Remote	0008.3b79.8787	3	OK
gi1	Bernie	03:08:58	Remote	0008.3b79.3232	9	OK
gi2	John	08:19:17	Remote	0008.3b89.3127	2	
gi3	Paul	02:12:48	Remote	0008.3b89.8237	8	Warning

```
Switch# show dot1x users username Bob
```

Port	Username	Session Time	Auth Method	MAC Address	VLAN	Filter
gi1	Bob	1d 09:07:38	Remote	0008.3b79.8787	3	OK

Filter ID #1: Supplicant-IPv4  
Filter ID #2: Supplicant-IPv6

**20.13 show dot1x statistics**

Use the **show dot1x statistics** Privileged EXEC mode command to display 802.1x statistics for the specified interface.

### Syntax

**show dot1x statistics interface** *interface-id*

### Parameters

**interface-id**—Specifies an interface ID. The interface ID must be an Ethernet port.

### Command Mode

Privileged EXEC mode

### Example

The following example displays 802.1x statistics for `gi 1`.

---

```
Console# show dot1x statistics interface gi1
EapolFramesRx: 11
EapolFramesTx: 12
EapolStartFramesRx: 1
EapolLogoffFramesRx: 1
EapolRespIdFramesRx: 3
EapolRespFramesRx: 6
EapolReqIdFramesTx: 3
EapolReqFramesTx: 6
InvalidEapolFramesRx: 0
EapLengthErrorFramesRx: 0
LastEapolFrameVersion: 1
LastEapolFrameSource: 00:08:78:32:98:78
```



The following table describes the significant fields shown in the display:

Field	Description
<b>EapolFramesRx</b>	The number of valid EAPOL frames of any type that have been received by this Authenticator.
<b>EapolFramesTx</b>	The number of EAPOL frames of any type that have been transmitted by this Authenticator.
<b>EapolStartFramesRx</b>	The number of EAPOL Start frames that have been received by this Authenticator.
<b>EapolLogoffFramesRx</b>	The number of EAPOL Logoff frames that have been received by this Authenticator.
<b>EapolRespIdFramesRx</b>	The number of EAP Resp/Id frames that have been received by this Authenticator.
<b>EapolRespFramesRx</b>	The number of valid EAP Response frames (other than Resp/Id frames) that have been received by this Authenticator.
<b>EapolReqIdFramesTx</b>	The number of EAP Req/Id frames that have been transmitted by this Authenticator.
<b>EapolReqFramesTx</b>	The number of EAP Request frames (other than Req/Id frames) that have been transmitted by this Authenticator.
<b>InvalidEapolFramesRx</b>	The number of EAPOL frames that have been received by this Authenticator for which the frame type is not recognized.
<b>EapLengthErrorFramesRx</b>	The number of EAPOL frames that have been received by this Authenticator in which the Packet Body Length field is invalid.
<b>LastEapolFrameVersion</b>	The protocol version number carried in the most recently received EAPOL frame.
<b>LastEapolFrameSource</b>	The source MAC address carried in the most recently received EAPOL frame.

---

## 20.14 clear dot1x statistics

Use the **clear dot1x statistics** Privileged EXEC mode command to clear 802.1x statistics.

### Syntax

**clear dot1x statistics** [*interface-id*]

### Parameters

*interface-id*—Specifies an interface ID. The interface ID must be an Ethernet port.

### Default Configuration

N/A

### Command Mode

Privileged EXEC

### User Guidelines

The command clears the statistics displayed in the **show dot1x statistics** command

### Example

---

```
console# clear dot1x statistics
```

---

## 20.15 dot1x host-mode

Use the **dot1x host-mode** Interface Configuration mode command to allow a single host (client) or multiple hosts on an IEEE 802.1x-authorized port. Use the **no** form of this command to return to the default setting.

### Syntax

**dot1x host-mode** {*multi-host* / *single-host* / *multi-sessions*}

### Parameters

- **multi-host**—Enable multiple-hosts mode.
- **single-host**—Enable single-hosts mode.

- **multi-sessions**—Enable multiple-sessions mode.

### Default Configuration

Default mode is multi-host.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

In multiple hosts mode only one of the attached hosts must be successfully authorized for all hosts to be granted network access. If the port becomes unauthorized, all attached clients are denied access to the network.

In multiple sessions mode each host must be successfully authorized in order to grant network access. Please note that packets are NOT encrypted, and after success full authentication filtering is based on the source MAC address only.

Port security on a port cannot be enabled in single-host mode and in multiple-sessions mode.

It is recommended to enable reauthentication when working in multiple-sessions mode in order to detect User Logout for users that hadn't sent Logoff.

### Example

```
console(config)# interface gil
console(config-if)# dot1x host-mode multi-host
console(config-if)# dot1x host-mode single-host
console(config-if)# dot1x host-mode multi-sessions
```

---

## 20.16 dot1x violation-mode

Use the **dot1x violation-mode** Interface Configuration (Ethernet) mode command to configure the action to be taken, when a station whose MAC address is not the supplicant MAC address, attempts to access the interface. Use the **no** form of this command to return to default.

### Syntax

```
dot1x violation-mode {restrict / protect / shutdown} [traps seconds]
```

```
no dot1x violation-mode
```

### Parameters

- **restrict**—Generates a trap when a station whose MAC address is not the supplicant MAC address, attempts to access the interface. The minimum time between the traps is 1 second. Those frames are forwarded but their source address are not learned.
- **protect**—Discard frames with source addresses not the supplicant address.
- **shutdown**—Discard frames with source addresses not the supplicant address and shutdown the port
- **trap *seconds*** - Send SNMP traps, and specifies the minimum time between consecutive traps. If seconds = 0 traps are disabled. If the parameter is not specified it defaults to 1 second for the restrict mode and 0 for the other modes.

### Default Configuration

Protect

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

The command is relevant only for single-host mode.

BPDU message whose MAC address is not the supplicant MAC address wouldn't be discarded in the protect mode.

BPDU message whose MAC address is not the supplicant MAC address would cause a shutdown in the shutdown mode.

### Example

---

```
console(config)# interface gil
console(config-if)# dot1x violation-mode protect
```

---

## 20.17 dot1x guest-vlan

Use the **dot1x guest-vlan** Interface Configuration (VLAN) mode command to define a guest VLAN. Use the **no** form of this command to restore the default configuration.

### Syntax

**dot1x guest-vlan**

**no dot1x guest-vlan**

### Default Configuration

No VLAN is defined as a guest VLAN.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

Use the **dot1x guest-vlan enable** Interface Configuration mode command to enable unauthorized users on an interface to access the guest VLAN.

If the guest VLAN is defined and enabled, the port automatically joins the guest VLAN when the port is unauthorized and leaves it when the port becomes authorized. To be able to join or leave the guest VLAN, the port should not be a static member of the guest VLAN.

### Example

The following example defines VLAN 2 as a guest VLAN.

---

```
Console(config)# interface vlan 2
Console(config-if)# dot1x guest-vlan
```

---

## 20.18 dot1x guest-vlan timeout

Use the **dot1x guest-vlan timeout** Global Configuration mode command to set the time delay between enabling 802.1x (or port up) and adding a port to the guest VLAN. Use the **no** form of this command to restore the default configuration.

### Syntax

**dot1x guest-vlan timeout** *timeout*

**no dot1x guest-vlan timeout**

### Parameters

**timeout**—Specifies the time delay in seconds between enabling 802.1x (or port up) and adding the port to the guest VLAN. (Range: 30–180)

### Default Configuration

The guest VLAN is applied immediately.

### Command Mode

Global Configuration mode

### User Guidelines

This command is relevant if the guest VLAN is enabled on the port. Configuring the timeout adds delay from enabling 802.1x (or port up) to the time the device adds the port to the guest VLAN.

### Example

The following example sets the delay between enabling 802.1x and adding a port to a guest VLAN to 60 seconds.

---

```
Console(config)# dot1x guest-vlan timeout 60
```

---

## 20.19 dot1x guest-vlan enable

Use the **dot1x guest-vlan enable** Interface Configuration (Ethernet) mode command to enable unauthorized users on the interface access to the guest VLAN. Use the **no** form of this command to disable access.

### Syntax

**dot1x guest-vlan enable**

**no dot1x guest-vlan enable**

### Default Configuration

The default configuration is disabled.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

A device can have only one global guest VLAN. The guest VLAN is defined using the **dot1x guest-vlan** Interface Configuration mode command.

### Example

The following example enables unauthorized users on `gi 1` to access the guest VLAN.

---

```
Console(config)# interface gi15
Console(config-if)# dot1x guest-vlan enable
```

---

## 20.20 dot1x mac-authentication

Use the **dot1x mac-authentication** Interface Configuration (Ethernet) mode command to enable authentication based on the station's MAC address. Use the **no** form of this command to disable access.

### Syntax

```
dot1x mac-authentication {mac-only | mac-and-802.1x}
```

```
no dot1x mac-authentication
```

### Parameters

- **mac-only**—Enables authentication based on the station's MAC address only. 802.1X frames are ignored.
- **mac-and-802.1x**—Enables 802.1X authentication and MAC address authentication on the interface.

### Default Configuration

Authentication based on the station's MAC address is disabled.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

The guest VLAN must be enabled when MAC authentication is enabled.

Static MAC addresses cannot be authorized. Do not change an authenticated MAC address to a static address.

It is not recommended to delete authenticated MAC addresses.

Reauthentication must be enabled when working in this mode.

### Example

The following example enables authentication based on the station's MAC address on `gi1`.

---

```
Console(config)# interface gi1
Console(config-if)# dot1x mac-authentication mac-only
```

---

## 20.21 show dot1x advanced

Use the **show dot1x advanced** Privileged EXEC mode command to display 802.1x advanced features for the device or specified interface.

### Syntax

```
show dot1x advanced [interface-id]
```

### Parameters

**interface-id**—Specify an interface ID. The interface ID must be an Ethernet port.

### Command Mode

Privileged EXEC mode

### Example

The following example displays 802.1x advanced features for the device.



---

```
console# show dot1x advanced
```

```
Guest VLAN: 3978
```

```
Guest VLAN Timeout:
```

```
Unauthenticated VLANs: 91, 92
```

```
Interface Multiple Guest   MAC                               Hosts   VLAN
Authentication -----
Enabled MAC-and-802.1X gi2   Enabled Disabled Disabled
```

```
Switch# show dot1x advanced gil
```

```
Interface Multiple Guest   MAC                               Hosts   VLAN
Authentication
```

```
----- gil Disabled Enabled
```

```
MAC-and-802.1X Legacy-Supp mode is disabled
```

```
Policy assignment resource err handling: Accept
```

```
Single host parameters
```

```
Violation action: Discard
```

```
Trap: Enabledx
```

```
Status: Single-host locked
```

```
Violations since last trap: 9
```

---

# 21 Ethernet Configuration Commands

---

## 21.1 interface

Use the **interface** Global Configuration mode command to configure an interface and enter interface configuration mode.

### Syntax

**interface** *interface-id*

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

### Example - For Gigabit Ethernet ports:

---

```
console(config)# interface gil-20
```

### Example - For Fast Ethernet ports:

---

```
console(config)# interface fa1-2
```

### Example - For Port Channels (LAGs):

---

```
console(config)# interface port-channel 1
```

---

## 21.2 interface range

Use the **interface range** command to execute a command on multiple ports at the same time.

### Syntax

**interface range** *interface-id-list*

### Parameters

**interface-id-list**—Specify list of interface IDs. The interface ID can be one of the following types: Ethernet port or Port-channel

### User Guidelines

Commands under the interface range context are executed independently on each interface in the range: If the command returns an error on one of the interfaces, it does not stop the execution of the command on other interfaces.

### Example

```
console(config)# interface range gi1-20
```

---

## 21.3 shutdown

Use the **shutdown** Interface Configuration (Ethernet, Port-channel) mode command to disable an interface. Use the **no** form of this command to restart a disabled interface.

### Syntax

**shutdown**

**no shutdown**

### Default Configuration

The interface is enabled.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### Example

**Example 1** - The following example disables `gi5` operations.

---

```
Console(config)# interface gi5
```

```
Console(config-if)# shutdown
```

```
Console(config-if)#
```

**Example 2** - The following example restarts the disabled Ethernet port.

---

```
Console(config)# interface gi5
```

```
Console(config-if)# no shutdown
```

```
Console(config-if)
```

---

## 21.4 description

Use the **description** Interface Configuration (Ethernet, Port-channel) mode command to add a description to an interface. Use the **no** form of this command to remove the description.

### Syntax

**description** *string*

**no description**

### Parameters

**string**—Specifies a comment or a description of the port to assist the user. (Length: 1–64 characters)

### Default Configuration

The interface does not have a description.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### Example

The following example adds the description 'SW#3' to gi5.

---

```
Console(config)# interface gi5
Console(config-if)# description SW#3
```

---

## 21.5 speed

Use the **speed** Interface Configuration (Ethernet, Port-channel) mode command to configure the speed of a given Ethernet interface when not using auto-negotiation. Use the **no** form of this command to restore the default configuration.

### Syntax

**speed** {10 | 100 | 1000}

## no speed

### Parameters

- **10**—Forces 10 Mbps operation.
- **100**—Forces 100 Mbps operation.
- **1000**—Forces 1000 Mbps operation.

### Default Configuration

The port operates at its maximum speed capability.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

The **no speed** command in a Port-channel context returns each port in the Port-channel to its maximum capability.

### Example

The following example configures the speed of `gi5` to 100 Mbps operation.

```
Console(config)# interface gi5
Console(config-if)# speed 100
```

---

## 21.6 duplex

Use the **duplex** Interface Configuration (Ethernet, Port-channel) mode command to configure the full/half duplex operation of a given Ethernet interface when not using auto-negotiation. Use the **no** form of this command to restore the default configuration.

### Syntax

**duplex** {*half* / *full*}

**no duplex**

### Parameters

- **half**—Forces half-duplex operation.

- **full**—Forces full-duplex operation.

### Default Configuration

The interface operates in full duplex mode.

### Command Mode

Interface Configuration (Port-channel) mode

### Example

The following example configures `gi5` to operate in full duplex mode.

---

```
Console(config)# interface gi5
Console(config-if)# duplex full
```

---

## 21.7 negotiation

Use the **negotiation** Interface Configuration (Ethernet, Port-channel) mode command to enable auto-negotiation operation for the speed and duplex parameters of a given interface. Use the **no** form of this command to disable auto-negotiation.

### Syntax

```
negotiation [capability [capability2 ... capability5]]
```

```
no negotiation
```

### Parameters

**capability**—Specifies the capabilities to advertise. (Possible values: 10h, 10f, 100h, 100f, 1000f). If unspecified, defaults to list of all the capabilities of the port.

### Default Configuration

Auto-negotiation is enabled.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### Example

The following example enables auto-negotiation on `gi5`.

---

```
Console(config)# interface gi5
Console(config-if)# negotiation
```

---

## 21.8 flowcontrol

Use the **flowcontrol** Interface Configuration (Ethernet, Port-channel) mode command to configure the flow control on a given interface. Use the **no** form of this command to disable flow control.

### Syntax

**flowcontrol** *{auto / on / off}*

**no flowcontrol**

### Parameters

- **auto**—Specifies auto-negotiation.
- **on**—Enables flow control.
- **off**—Disables flow control.

### Default Configuration

Flow control is disabled.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

Use the **negotiation** command to enable **flow control auto**.

### Example

The following example enables flow control on port `gi1`.

---

```
Console(config)# interface gi1
```

---

```
Console(config-if)# flowcontrol on
```

---

## 21.9 mdix

Use the **mdix** Interface Configuration (Ethernet) mode command to enable cable crossover on a given interface. Use the **no** form of this command to disable cable crossover.

### Syntax

```
mdix {on / auto}
```

```
no mdix
```

### Parameters

- **on**—Enables manual MDIX.
- **auto**—Enables automatic MDI/MDIX.

### Default Configuration

The default setting is On.

### Command Mode

Interface Configuration (Ethernet) mode

### Example

The following example enables automatic crossover on port `gi5`.

---

```
Console(config)# interface gi5
```

```
Console(config-if)# mdix auto
```

---

## 21.10 back-pressure

Use the **back-pressure** Interface Configuration (Ethernet) mode command to enable back pressure on a specific interface. Use the **no** form of this command to disable back pressure.



**Syntax****back-pressure****no back-pressure****Default Configuration**

Back pressure is disabled.

**Command Mode**

Interface Configuration (Ethernet) mode

**Example**The following example enables back pressure on port `gi5`.

---

```
Console(config)# interface gi5
Console(config-if)# back-pressure
```

---

**21.11 port jumbo-frame**Use the **port jumbo-frame** Global Configuration mode command to enable jumbo frames on the device. Use the **no** form of this command to disable jumbo frames.**Syntax****port jumbo-frame****no port jumbo-frame****Default Configuration**

Jumbo frames are disabled on the device.

**Command Mode**

Global Configuration mode

**User Guidelines**

This command takes effect only after resetting the device.

### Example

The following example enables jumbo frames on the device.

---

```
Console(config)# port jumbo-frame
```

---

## 21.12 clear counters

Use the **show interfaces counters** EXEC mode command to display traffic seen by all the physical interfaces or by a specific interface.

### Syntax

**show interfaces counters** *[interface-id]* *[detailed]*

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**detailed**—Displays information for non-present ports in addition to present ports.

### Command Mode

EXEC mode

### Example

The following example clears the statistics counters for gi5.

---

```
Console# clear counters gi5.
```

---

## 21.13 set interface active

Use the **set interface active** EXEC mode command to reactivate an interface that was shut down.

### Syntax

**set interface active** *{interface-id}*

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

### Command Mode

EXEC mode

### User Guidelines

This command is used to activate interfaces that were configured to be active, but were shut down by the system.

### Example

The following example reactivates `gi 1`.

---

```
Console# set interface active gi 1
```

---

## 21.14 show interfaces configuration

Use the **show interfaces configuration** EXEC mode command to display the configuration for all configured interfaces or for a specific interface.

### Syntax

```
show interfaces configuration [interface-id]
```

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

### Command Mode

EXEC mode

### Example

The following example displays the configuration of all configured interfaces:

```
console# show interfaces configuration
```

Port	Type	Duplex	Speed	Neg	Flow control	Admin State	Back Pressure	Mdix Mode
-----	-----	-----	-----	-----	-----	-----	-----	-----
gi1	1G-Copper	Full	10000	Disabled	Off	Up	Disabled	Off
gi2	1G-Copper	Full	1000	Disabled	Off	Up	Disabled	Off

PO	Type	Speed	Neg	Flow Control	Admin State
-----	-----	-----	-----	-----	-----
Po1			Disabled	Off	Up

## 21.15 show interfaces status

Use the **show interfaces status** EXEC mode command to display the status of all configured interfaces or of a specific interface.

### Syntax

**show interfaces status** [*interface-id*] [*detailed*]

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**detailed**—Displays information for non-present ports in addition to present ports.

### Command Mode

EXEC mode

### Example

The following example displays the status of all configured interfaces.

```
console# show interfaces status
```

Flow	Link	Back	Mdix
------	------	------	------

```

Port      Type      Duplex Speed Neg      ctrl State Pressure Mode
-----
gi1      1G-Copper Full   1000 Disabled Off   Up   Disabled Off
gi2      1G-Copper --     --     --     --   Down --     --
                                     Flow Link
Po1      Type      Duplex Speed Neg      ctrl State
-----
Po1      1G        Full   10000 Disabled Off   Up

```

## 21.16 show interfaces advertise

Use the **show interfaces advertise** EXEC mode command to display auto-negotiation advertisement information for all configured interfaces or for a specific interface.

### Syntax

**show interfaces advertise** *[interface-id]*

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

### Command Mode

EXEC mode

### Examples

The following examples display auto-negotiation information.

```

Console# show interfaces advertise
Port      Type      Neg      Operational Link
-----
gi1      1G-Copper Enabl    1000f, 100f, 10f, 10h
gi2      1G-Copper Enabl    1000f
                                     e

```

```

Console# show interfaces advertise gi1

Port:gil
Type: 1G-Copper
Link state: Up
Auto Negotiation: enabled

                               10h   10f   100   100f  1000f
                               ---   ---   h     ----  -----
Admin Local link               yes   yes   -     yes   yes
Advertisement
Oper Local link                yes   yes   yes   yes   yes
Advertisement
Remote Local link              -     -     yes   -     yes
Advertisement
Priority Resolution              -

```

```

Console# show interfaces advertise gi1

Port: gil
Type: 1G-Copper
Link state: Up
Auto negotiation: disabled.

```

## 21.17 show interfaces description

Use the **show interfaces description** EXEC mode command to display the description for all configured interfaces or for a specific interface.

### Syntax

**show interfaces description** *[interface-id]*

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Command Mode**

EXEC mode

**Example**

The following example displays the description of all configured interfaces.

---

```

Console# show interfaces description
Port      Descriptions
-----
gi1       -----
gi2       Port that should be used for management only
gi3
gi4

Po        Description
-----
Po1       Output

```

**21.18 show interfaces counters**

Use the **show interfaces counters** EXEC mode command to display traffic seen by all the physical interfaces or by a specific interface.

**Syntax**

```
show interfaces counters [interface-id]
```

**Parameters**

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Command Mode**

EXEC mode

**Example**

The following example displays traffic seen by all the physical interfaces.

```
console# show interfaces counters gil

Port          InUcastPkts  InMcastPkts  InBcastPkts  InOctets
-----
gil           0             0             0             0
Port          OutUcastPkts OutMcastPkts OutBcastPkts OutOctets
-----
          gil      0             1             35            7051

Alignment Errors: 0
FCS Errors: 0
Single Collision Frames: 0
Multiple Collision Frames: 0
SQE Test Errors: 0
Deferred Transmissions: 0
Late Collisions: 0
Excessive Collisions: 0
Carrier Sense Errors: 0
Oversize Packets: 0
Internal MAC Rx Errors: 0
Symbol Errors: 0
Received Pause Frames: 0
Transmitted Pause Frames: 0
```



The following table describes the fields shown in the display.

Field	Description
<b>InOctets</b>	The number of received octets.
<b>InUcastPkts</b>	The number of received unicast packets.
<b>InMcastPkts</b>	The number of received multicast packets.
<b>InBcastPkts</b>	The number of received broadcast packets.
<b>OutOctets</b>	The number of transmitted octets.
<b>OutUcastPkts</b>	The number of transmitted unicast packets.
<b>OutMcastPkts</b>	The number of transmitted multicast packets.
<b>OutBcastPkts</b>	The number of transmitted broadcast packets.
<b>FCS Errors</b>	The number of frames received that are an integral number of octets in length but do not pass the FCS check.
<b>Single Collision Frames</b>	The number of frames that are involved in a single collision, and are subsequently transmitted successfully.
<b>Multiple Collision Frames</b>	The number of frames that are involved in more than one collision and are subsequently transmitted successfully.
<b>SQE Test Errors</b>	The number of times that the SQE TEST ERROR is received. The SQE TEST ERROR is set in accordance with the rules for verification of the SQE detection mechanism in the PLS Carrier Sense Function as described in IEEE Std. 802.3, 2000 Edition, section 7.2.4.6.
<b>Deferred Transmissions</b>	The number of frames for which the first transmission attempt is delayed because the medium is busy.
<b>Late Collisions</b>	The number of times that a collision is detected later than one slotTime into the transmission of a packet.
<b>Excessive Collisions</b>	The number of frames for which transmission fails due to excessive collisions.

---

Field	Description
<b>Oversize Packets</b>	The number of frames received that exceed the maximum permitted frame size.
<b>Internal MAC Rx Errors</b>	The number of frames for which reception fails due to an internal MAC sublayer receive error.
<b>Received Pause Frames</b>	The number of MAC Control frames received with an opcode indicating the PAUSE operation.
<b>Transmitted Pause Frames</b>	The number of MAC Control frames transmitted on this interface with an opcode indicating the PAUSE operation.

---

### 21.19 show port jumbo-frame

Use the **show port jumbo-frame** EXEC mode command to display the configuration of jumbo frames.

#### Syntax

**show port jumbo-frame**

#### Command Mode

EXEC mode

#### Example

The following example displays the configuration of jumbo frames on the device.

---

```
Console# show port jumbo-frame
Jumbo frames are disabled
Jumbo frames will be enabled after reset
```

---

## 21.20 storm-control broadcast level kbps

Use the **storm-control broadcast level** Interface Configuration mode command to configure the maximum rate of broadcast on a port. Use the **no** form of this command to return to default.

### Syntax

**storm-control broadcast level kbps** *kbps*

**no storm-control broadcast level**

### Parameters

**kbps**—Maximum of kilo bits per second of broadcast traffic on a port. (Range 70-1000000)

### Default Configuration

1000

### Command Mode

Interface Configuration mode (Ethernet)

### User Guidelines

Use the **storm-control broadcast enable** Interface Configuration command to enable storm control.

The calculated rate includes the 20 bytes of Ethernet framing overhead (preamble+SFD+IPG).

### Example

---

```
console(config)# interface gil
console(config-if)# storm-control broadcast level kbps 12345
```

---

## 21.21 storm-control broadcast level

Use the **storm-control broadcast level** Interface Configuration mode command to configure the maximum rate of broadcast. Use the **no** form of this command to return to default.

### Syntax

**storm-control broadcast level** {*level* | **kbps** *kbps*}

**no storm-control broadcast level**

### Parameters

**level** - Suppression level in percentage. Block the flooding of storm packets when the value specified for level is reached. (Range 1 -100)

**kbps**—Maximum of kilobits per second of broadcast traffic on a port. (Range 8500–10000000)

### Default Configuration

**level** - 10%

**kbps**—10% of port speed in Kbps

### Command Mode

Interface Configuration mode (Ethernet)

### User Guidelines

Use the **storm-control broadcast enable** Interface Configuration command to enable storm control.

The calculated rate includes the 20 bytes of Ethernet framing overhead (preamble+SFD+IPG).

The command **smartport storm-control broadcast level** has the same functionality except that it does not return an error (and does not configure anything) when executing it for a port-channel.

### Example

```
console(config)# interface gil
console(config-if)# storm-control broadcast level 50 kbps 12345
```

---

## 21.22 storm-control include-multicast

Use the **storm-control include-multicast** Interface Configuration mode command to count multicast packets in the broadcast storm control. Use the **no** form of this command to disable counting of multicast packets in the broadcast storm control.

### Syntax

**storm-control include-multicast** [*unknown-unicast*]

**no storm-control include-multicast**

### Parameters

N/A

**unknown-unicast**—Specifies also the count of unknown unicast packets.

### Default Configuration

Disabled

### Command Mode

Interface Configuration mode (Ethernet)

### User Guidelines

The command **smartport storm-control include-multicast** has the same functionality except that it does not return an error (and does not configure anything) when executing it for a port-channel

### Example

---

```
console(config)# interface gil
console(config-if)# storm-control include-multicast
```

---

## 21.23 show storm-control

Use the **show storm-control EXEC** mode command to display the configuration of storm control.

### Syntax

**show storm-control** [*interface-id*]

**Parameters**

**interface-id**—Specifies the interface.

**Command Mode**

EXEC mode

**Example**

```
console# show storm-control
```

Port	State	Admin Rate	Oper Rate	Included
			[Kb/Sec]	
-----	-----	-----	-----	-----
gi1	Enabled	12345 Kb/Sec	12345	Broadcast, Multicast, Unknown Unicast
gi2	Disabled	100000 Kb/Sec	100000	Broadcast
gi3	Enabled	10%	000000	Broadcast

## 22 PHY Diagnostics Commands

### 22.1 show cable-diagnostics cable-length

Use the **show cable-diagnostics cable-length** EXEC mode command to display the estimated copper cable length attached to all ports or to a specific port.

#### Syntax

**show cable-diagnostics cable-length** [*interface interface-id*]

#### Parameters

**interface-id**—Specify an interface ID. The interface ID must be an Ethernet port.

#### Command Mode

EXEC mode

#### User Guidelines

The port must be active and working at 100 M or 1000 M.

#### Example

The following example displays the estimated copper cable length attached to all ports.

```
Console> show cable-diagnostics cable-length
Port          Length [meters]
-----
gi1           < 50
gi2           Copper not active
gi3           110-140
gi4           Fiber
```

### 22.2 show fiber-ports optical-transceiver

Use the **show fiber-ports optical-transceiver** EXEC mode command to display the optical transceiver diagnostics.

**Syntax**

**show fiber-ports optical-transceiver** [*interface interface-id*] [*detailed*]

**Parameters**

- **interface-id**—Specifies an interface ID. The interface ID must be an Ethernet port.
- **detailed**—Displays detailed diagnostics.

**Command Mode**

EXEC mode

**Example**

The following examples display the optical transceiver diagnostics results.

---

```

console# show fiber-ports optical-transceiver

  Port      Temp  Voltage Current  Output Input  LOS
                Power  Power
-----
  gi1       W     OK     OK     OK     OK     OK
  gi2       OK    OK     OK     E     OK     OK

Temp        - Internally measured transceiver temperature
Voltage     - Internally measured supply voltage
Current     - Measured TX bias current
Output Power - Measured TX output power in milliWatts
Input Power - Measured RX received power in milliWatts
LOS         - Loss of signal

N/A - Not Available, N/S - Not Supported,
W - Warning, E - Error
console# show fiber-ports optical-transceiver detailed

```

```

  Port      Temp  Voltage Current  Output  Input  LOS
                [C]  [Volt]  [mA]   Power  Power
                [mWatt] [mWatt]

```



```
-----  
gi1      Copper  
gi6      Copper  
gi7      28      3.32      7.26      3.53      3.68      No  
gi8      29      3.33      6.50      3.53      3.71      No  
Temp          - Internally measured transceiver temperature  
Voltage       - Internally measured supply voltage  
Current       - Measured TX bias current  
Output Power  - Measured TX output power in milliWatts  
Input Power   - Measured RX received power in milliWatts  
LOS          - Loss of signal  
N/A - Not Available, N/S - Not Supported, W - Warning, E - Error
```

---

## 23 Power over Ethernet (PoE) Commands

### 23.1 power inline

Use the **power inline** Interface Configuration mode command to configure the inline power administrative mode on an interface.

#### Syntax

```
power inline {auto / never}
```

#### Parameters

- **auto**—Turns on the device discovery protocol and applies power to the device.
- **never**—Turns off the device discovery protocol and stops supplying power to the device.

#### Default Configuration

The default configuration is set to auto.

#### Command Mode

Interface Configuration (Ethernet) mode

#### Example

The following example turns on the device discovery protocol on port 4.

```
Console(config)# interface gi4  
Console(config-if)# power inline auto
```

### 23.2 power inline priority

Use the **power inline priority** Interface Configuration (Ethernet) mode command to configure the interface inline power management priority. Use the **no** form of this command to restore the default configuration.

#### Syntax

```
power inline priority {critical / high / low}
```

## no power inline priority

### Parameters

- **critical**—Specifies that the powered device operation is critical.
- **high**—Specifies that the powered device operation is high priority.
- **low**—Specifies that the powered device operation is low priority.

### Default Configuration

The default configuration is set to low priority.

### Command Mode

Interface Configuration (Ethernet) mode

### Example

The following example sets the inline power management priority of gigabitethernet port 4 to High.

```
Console(config)# interface gi4
Console(config-if)# power inline priority high
```

## 23.3 power inline usage-threshold

Use the **power inline usage-threshold** Global Configuration mode command to configure the threshold for initiating inline power usage alarms. Use the **no** form of this command to restore the default configuration.

### Syntax

**power inline usage-threshold** *percent*

**no power inline usage-threshold**

### Parameters

**percent**—Specifies the threshold in percent to compare to the measured power. (Range: 1–99)

### Default Configuration

The default threshold is 95 percent.

### Command Mode

Global Configuration mode

### Example

The following example configures the threshold for initiating inline power usage alarms to 90 percent.

---

```
Console(config)# power inline usage-threshold 90
```

---

## 23.4 power inline traps enable

Use the **power inline traps enable** Global Configuration mode command to enable inline power traps. Use the **no** form of this command to disable traps.

### Syntax

**power inline traps enable**

**no power inline traps enable**

### Default Configuration

Inline power traps are disabled.

### Command Mode

Global Configuration mode

### Example

The following example enables inline power traps.

---

```
Console(config)# power inline traps enable
```

---

## 23.5 power inline limit

Use the **power inline limit** Interface Configuration mode command to configure the power limit per port on an interface. Use the **no** form of the command to return to default.

**Syntax**

**power inline limit** *power*

**no power inline limit**

**Parameters**

**power**—States the port power consumption limit in Milliwatts (Range: 0-15400)

**Default Configuration**

The default value is the maximum power allowed in the specific working mode:

- 15.4W

**Command Mode**

Interface Configuration (Ethernet) mode

**Example**

The following example sets inline power on a port.

---

```
console(config)# interface gi1
console(config-if)# power inline limit 2222
```

---

## 23.6 power inline limit-mode

Use the **power inline limit-mode** Global Configuration mode command to set the power limit mode of the system. Use the **no** form of this command to return to default.

**Syntax**

**power inline limit-mode** *{class / port}*

**no power inline limit-mode**

**Parameters**

- **class**—The power limit of a port is based on the class of the PD (Power Device) as detected during the classification process
- **port**—The power limit of a port is fixed regardless of the class of the discovered PD.

**Command Mode**

Global Configuration mode

**Example**

The following example sets the power limit to class.

---

```
console(config)# power inline limit-mode class
```

---

**23.7 show power inline**Use the **show power inline** EXEC mode command to display information about the inline power for all interfaces or for a specific interface.**Syntax****show power inline** [*interface-id*]**Parameters****interface-id**—Specifies an interface ID. The interface ID must be an Ethernet port.**Default Configuration**

There is no default configuration for this command.

**Command Mode**

EXEC mode

**Example**

The following example displays information about the inline power for all ports (port power based).

---

```
console(config)# show power inline

Port based power-limit mode

Unit  Power  Nominal  Consumed  Usage  Traps
      Power           Power      Threshold
-----
1     Off    1 Watts  0 Watts (0%)  95     Disable
```

---

```

Port      Powered Device      State      Status      Priority      Class
-----  -
gi1      IP Phone Model A    Auto      On          High         Class0
gi2      Wireless AP Model A Auto      On          Low          Class1
gi3                        Auto      Off         Low          N/A

```

### Example

The following example displays information about the inline power for a specific port.

```
console(config)# show power inline
```

```

Port      Powered Device      State      Status      Priority      Class
-----  -
gi1      IP Phone Model A    Auto      On          High         Class0
Power limit: 15 W
Power limit (for port power-limit mode): 15 W

```

The following table describes the fields shown in the display:

Field	Description
<b>Power</b>	The inline power sourcing equipment operational status.
<b>Nominal Power</b>	The inline power sourcing equipment nominal power in Watts.
<b>Consumed Power</b>	The measured usage power in Watts.
<b>Usage Threshold</b>	The usage threshold expressed in percent for comparing the measured power and initiating an alarm if threshold is exceeded.
<b>Traps</b>	Indicates if inline power traps are enabled.
<b>Port</b>	The Ethernet port number.
<b>Powered device</b>	A description of the powered device type.
<b>State</b>	Indicates if the port is enabled to provide power. The possible values are Auto or Never.

Field	Description
<b>Priority</b>	The port inline power management priority. The possible values are Critical, High or Low.
<b>Status</b>	Describes the port inline power operational state. The possible values are On, Off, Test-Fail, Testing, Searching or Fault.
<b>Class</b>	The power consumption classification of the powered device.
<b>Overload Counter</b>	Counts the number of overload conditions detected.
<b>Short Counter</b>	Counts the number of short conditions detected.
<b>Denied Counter</b>	Counts the number of times power was denied.
<b>Absent Counter</b>	Counts the number of times power was removed because powered device dropout was detected.
<b>Invalid Signature Counter</b>	Counts the number of times an invalid signature of a powered device was detected.

Following is a list of port status values:

Port is off - Underload disconnect detected

Port is off - Overload detected

Port is off - Short detected

Port is off - Invalid PD resistor signature detected

Port is on - Valid PD resistor signature detected

Port is off - Power was denied

Port is on - Valid capacitor signature detected

Port is off - Backoff state has occurred

Port is off - Class error has occurred

## 23.8 show power inline consumption

Use the **show power inline consumption** EXEC mode command to display information about the inline power consumption for all interfaces or for a specific interface.



**Syntax**

**show power inline consumption** *[interface-id]*

**Parameters**

**Interface-id**—Specifies an interface ID. The interface ID must be an Ethernet port.

**Default Configuration**

There is no default configuration for this command.

**Command Mode**

EXEC mode

**Example**

The following example displays information about the inline power consumption.

```
Console# show power inline consumption
```

Port	Power Limit(W)	Power (W)	Voltage(V)	Current(mA)
gi1	15.4	4.115	50.8	81
gi2	15.4	4.157	50.7	82
gi3	15.4	4.021	50.9	79

---

## 24 EEE Commands

---

### 24.1 eee enable (global)

Use the **eee enable** Global Configuration command to enable the EEE mode globally. Use the **no** format of the command to disable the mode.

#### Syntax

**eee enable**

**no eee enable**

#### Default Configuration

EEE is enabled.

#### Command Mode

Global Configuration mode

#### User Guidelines

In order for EEE to work, the device at the other end of the link must also support EEE and have it enabled. In addition, for EEE to work properly, Auto-Negotiation must be enabled; however, if the port speed is negotiated 1Giga, EEE always works regardless of the auto-negotiation status (meaning enable or disable).

If Auto-Negotiation is not enabled on the port and its speed is less than 1 Giga, the EEE Operational status is disabled.

---

### 24.2 eee enable (interface)

Use the **eee enable** Interface Configuration command to enable the EEE mode on an Ethernet port. Use the **no** format of the command to disable the mode.

#### Syntax

**eee enable**

**no eee enable**

#### Parameters

N/A

### Default Configuration

EEE is enabled.

### Command Mode

Interface Configuration mode (Ethernet)

### User Guidelines

If Auto-Negotiation is not enabled on the port and its speed is 1 Giga, the EEE Operational status is disabled.

---

## 24.3 eee lldp enable

Use the **eee lldp enable** Interface Configuration command to enable EEE support by LLDP on an Ethernet port. Use the **no** format of the command to disable the support.

### Syntax

**eee lldp enable**

**no eee lldp enable**

### Parameters

N/A

### Default Configuration

Enabled

### Command Mode

Interface Configuration mode (Ethernet)

### User Guidelines

Enabling EEE LLDP advertisement allows devices to choose and change system wake-up times in order to get the optimal energy saving mode.

---

## 24.4 show eee

Use the **show eee** EXEC command to display EEE information.

**Syntax**

```
show eee [interface-id]
```

**Parameters**

**interface-id**—Specify an interface ID. The interface ID must be an Ethernet port.

**Defaults**

N/A

**Command Mode**

EXEC

**Examples****Example 1 - Brief Information about all ports**

```
Switch>show eee
EEE globally enabled
EEE Administrative status is enabled on ports: gi1-6, gi7
EEE Operational status is enabled on ports: gi1, gi3-6, gi2, gi5
EEE LLDP Administrative status is enabled on ports: gi1-5
EEE LLDP Operational status is enabled on ports: gi1-5
```

**Example 2 - Port in state notPresent, no information if port supports EEE**

```
Switch> show eee gi10
Port Status: notPresent
EEE Administrative status: enabled
EEE LLDP Administrative status: enabled
EEE LLDP Administrative status: enabled
```

**Example 3 - Port in status DOWN**

```
Switch>show eee gi10
Port Status: DOWN
EEE capabilities:
```

```
Speed 10M: EEE not supported
Speed 100M: EEE supported
Speed 1G: EEE supported
EEE Administrate status: enabled
EEE LLDP Administrate status: enabled
```

---

#### Example 4 - Port in status UP and does not support EEE

```
Switch>show eee gi2
Port Status: UP
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
Current port speed: 1Gbps
EEE Administrate status: enabled
EEE LLDP Administrate status: enabled
```

---

#### Example 5 - Neighbor does not support EEE

```
Switch>show eee gi5
Port Status: UP
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
Current port speed: 1Gbps
EEE Remote status: disabled
EEE Administrate status: enabled
EEE Operational status: disabled (neighbor does not support)
EEE LLDP Administrate status: enabled
EEE LLDP Operational status: disabled
```

---

#### Example 6 - EEE is disabled on the port

```
Switch>show eee gi1
Port Status: UP
```

```
EEE capabilities:
  Speed 10M: EEE not supported
  Speed 100M: EEE supported
  Speed 1G: EEE supported
Current port speed: 1Gbps
EEE Administrate status: disabled
EEE Operational status: disabled
EEE LLDP Administrate status: enabled
EEE LLDP Operational status: disabled
```

---

### Example 7 - EEE is running on the port, EEE LLDP is disabled

```
Switch>show eee gi2
Port Status: UP
EEE capabilities:
  Speed 10M: EEE not supported
  Speed 100M: EEE supported
  Speed 1G: EEE supported
Current port speed: 1Gbps
EEE Remote status: enabled
EEE Administrate status: enabled
EEE Operational status: enabled
EEE LLDP Administrate status: disabled
EEE LLDP Operational status: disabled
Resolved Tx Timer: 10usec
Local Tx Timer: 10 usec
Resolved Timer: 25 usec
Local Rx Timer: 20 usec
```

---

### Example 8 - EEE and EEE LLDP are running on the port

```
Switch>show eee gi3
Port Status: UP
EEE capabilities:
  Speed 10M: EEE not supported
```

```
Speed 100M: EEE supported
Speed 1G: EEE supported
Current port speed: 1Gbps
EEE Remote status: enabled
EEE Administrative status: enabled
EEE Operational status: enabled
EEE LLDP Administrative status: enabled
EEE LLDP Operational status: enabled
Resolved Tx Timer: 10usec
Local Tx Timer: 10 usec
Remote Rx Timer: 5 usec
Resolved Timer: 25 usec
Local Rx Timer: 20 usec
Remote Tx Timer: 25 usec
```

---

**Example 9 - EEE is running on the port, EEE LLDP enabled but not synchronized with remote link partner**

```
Switch>show eee gi9
Port Status: up
EEE capabilities:
  Speed 10M: EEE not supported
  Speed 100M: EEE supported
  Speed 1G: EEE supported
Current port speed: 1Gbps
EEE Remote status: enabled
EEE Administrative status: enabled
EEE Operational status: enabled
EEE LLDP Administrative status: enabled
EEE LLDP Operational status: disabled
Resolved Tx Timer: 64
Local Tx Timer: 64
Resolved Rx Timer: 16
```

Local Rx Timer: 16

---

### Example 10 - EEE and EEE LLDP are running on the port

```
Switch>show eee gi3
Port Status: UP
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
Current port speed: 1Gbps
EEE Remote status: enabled
EEE Administrative status: enabled
EEE Operational status: enabled
EEE LLDP Administrative status: enabled
EEE LLDP Operational status: enabled
Resolved Tx Timer: 10usec
Local Tx Timer: 10 usec
Remote Rx Timer: 5 usec
Resolved Timer: 25 usec
Local Rx Timer: 20 usec
Remote Tx Timer: 25 usec
```



---

## 25 Green Ethernet

---

### 25.1 green-ethernet energy-detect (global)

Use the **green-ethernet energy-detect** Global Configuration mode command to enable Green-Ethernet Energy-Detect mode globally. Use the **no** form of this command to disabled it.

#### Syntax

**green-ethernet energy-detect**

**no green-ethernet energy-detect**

#### Parameters

N/A

#### Default Configuration

Enabled.

#### Command Mode

Global Configuration mode

#### Example

---

```
console(config)# green-ethernet energy-detect
```

---

### 25.2 green-ethernet energy-detect (interface)

Use the **green-ethernet energy-detect** Interface configuration mode command to enable green-ethernet Energy-Detect mode on an interface. use the no form of this command, to disable it on an interface

#### Syntax

**green-ethernet energy-detect**

**no green-ethernet energy-detect**

### Parameters

N/A

### Default Configuration

Enabled

### Command Mode

Interface configuration mode (Ethernet)

### User Guidelines

Energy-Detect can work only when the port is copper. When a port is enabled for auto selection copper/fiber Energy-Detect cannot work.

It takes the PHY ~5 seconds to fall into sleep mode when the link is lost after normal operation.

### Example

```
console(config)# interface gil
console(config-if)# green-ethernet energy-detect
```

## 25.3 show green-ethernet

Use the **show green-ethernet** Privileged EXEC mode command to show green-ethernet configuration and information.

### Syntax

```
show green-ethernet [interface-id]
```

### Parameters

*interface-id*—Specifies an interface ID. The interface ID must be an Ethernet port.

### Parameters Range

Default. When no interface is specified, this command shows information for all interfaces.

**Command Mode**

Privileged EXEC mode

**User Guidelines**

The following describes all possible reasons the **show** command displays, and their descriptions.

If there are a several reasons for non-operation, then only the highest priority reason is displayed.

Energy-detect Non-operational Reasons		
Priority	Reason	Description
1	NP	Port is not present
2	LT	Link Type is not supported (fiber, auto media select)
3	LU	Port Link is up – NA

Short-Reach Non-operational Reasons		
Priority	Reason	Description
1	NP	Port is not present
2	LT	Link Type is not supported (fiber)
3	LS	Link Speed Is not Supported (100M,10M,10G)
4	LL	Link Length received from VCT Test exceed threshold
6	LD	Port Link is Down – NA

**Example**

```

console# show green-ethernet

Energy-Detect mode: Enabled

Short-Reach mode: Disabled

Power Consumption: 76% (3.31W out of maximum 4.33W)

Cumulative Energy Saved: 33 [Watt*Hour]

Short-Reach cable length threshold: 50m

```

Port	Energy-Detect			Short-Reach			VCT Cable	
	Admin	Oper	Reason	Admin	Force	Oper	Reason	Length
gi1	on	on		off	off	off		
gi2	on	off	LU	on	off	off		< 50
gi3	on	off	LU	off	off	off		

## 25.4 green-ethernet short-reach (global)

Use the **green-ethernet short-reach** Global Configuration mode command to enable green-ethernet short-reach mode globally. Use the **no** form of this command to disabled it.

### Syntax

**green-ethernet short-reach**

**no green-ethernet short-reach**

### Parameters

N/A

### Default Configuration

Disabled.

### Command Mode

Global Configuration mode

### Example

```
console(config)# green-ethernet short-reach
```

## 25.5 green-ethernet short-reach (interface)

Use the **green-ethernet short-reach** Interface Configuration mode command to enable green-ethernet short-reach mode on an interface. Use the **no** form of this command to disable it on an interface.

### Syntax

**green-ethernet short-reach**

**no green-ethernet short-reach**

### Parameters

N/A

### Default Configuration

Disabled.

### Command Mode

Interface Configuration mode (Ethernet)

### User Guidelines

When **Short-Reach** mode is enabled and is not forced, the VCT (Virtual Cable Tester) length check must be performed. The VCT length check can be performed only on a copper port operating at a speed of 1000 Mbps. If the media is not copper or the link speed is not 1000 Mbps and short-reach mode is not forced (by **green-ethernet short-reach force**), Short-Reach mode is not applied.

When the interface is set to enhanced mode, after the VCT length check has completed and set the power to low, an active monitoring for errors is done continuously. In the case of errors crossing a certain threshold, the PHY will be reverted to long reach.

Note that EEE cannot be enabled if the Short-Reach mode is enabled.

### Example

---

```
console(config)# interface gil
```

```
console(config-if)# green-ethernet short-reach
```

---

---

## 25.6 green-ethernet power-meter reset

Use the **green-ethernet power meter reset** Privileged EXEC mode command to reset the power save meter.

**Syntax****green-ethernet power-meter reset****Command Mode**

Privileged EXEC mode.

**Example**

---

```
console(config)# green-ethernet power-meter reset
```

## 26 Port Channel Commands

Use the **channel-group** Interface Configuration (Ethernet) mode command to associate a port with a port-channel. Use the **no** form of this command to remove a port from a port-channel.

### Syntax

```
channel-group port-channel mode {on | auto}
```

```
no channel-group
```

### Parameters

- **port-channel**—Specifies the port channel number for the current port to join.
- **mode {on | auto}**—Specifies the mode of joining the port channel. The possible values are:
  - **on**—Forces the port to join a channel without an LACP operation.
  - **auto**—Forces the port to join a channel as a result of an LACP operation.

### Default Configuration

The port is not assigned to a port-channel.

### Command Mode

Interface Configuration (Ethernet) mode

### Example

The following example forces port `gi1` to join port-channel 1 without an LACP operation.

---

```
Console(config)# interface gi1
Console(config-if)# channel-group 1 mode on
```

---

## 26.1 port-channel load-balance

Use the **port-channel load-balance** Global Configuration mode command to configure the load balancing policy of the port channeling. Use the **no** form of this command to reset to default.

### Syntax

**port-channel load-balance** *{src-dst-mac src-dst-mac-ip }*

**no port-channel load-balance**

### Parameters

- **src-dst-mac**—Port channel load balancing is based on the source and destination MAC address.
- **src-dst-mac-ip**—Port channel load balancing is based on the source and destination of MAC and IP addresses.

### Default Configuration

src-dst-mac is the default option.

### Command Mode

Global Configuration mode

### User Guidelines

In **src-dst-mac-ip-port** load balancing policy, fragmented packets might be reordered.

### Example

---

```
console(config)# port-channel load-balance src-dst-mac
console(config)# port-channel load-balance src-dst-mac-ip
console(config)# port-channel load-balance src-dst-mac-ip-port
```

---

## 26.2 show interfaces port-channel

Use the **show interfaces port-channel** EXEC mode command to display port-channel information for all port channels or for a specific port channel.



### Syntax

**show interfaces port-channel** [*interface-id*]

### Parameters

**interface-id**—Specify an interface ID. The interface ID must be a Port Channel.

### Command Mode

EXEC mode

### Example

The following example displays information on all port-channels.

---

```
console# show interfaces port-channel

Load balancing: src-dst-mac.
Gathering information...

Channel  Ports
-----  -----
Po1      Active: gi1,Inactive: gi2-3
Po2      Active: gi5 Inactive: gi4

console# show interfaces switchport gi1

Gathering information...

Name: gi1
Switchport: enable
Administrative Mode: access
Operational Mode: down
Access Mode VLAN: 1
Access Multicast TV VLAN: none
Trunking Native Mode VLAN: 1
Trunking VLANs Enabled: 1
                               2-4094 (Inactive)
```

---

```
General PVID: 1
General VLANs Enabled: none
General Egress Tagged VLANs Enabled: none
General Forbidden VLANs: none
General Ingress Filtering: enabled
General Acceptable Frame Type: all
General GVRP status: disabled
Customer Mode VLAN: none
Private-vlan promiscuous-association primary VLAN: none
Private-vlan promiscuous-association Secondary VLANs Enabled: none
Private-vlan host-association primary VLAN: none
Private-vlan host-association Secondary VLAN Enabled: none
DVA: disable
```

---

---

## 27 Address Table Commands

---

### 27.1 bridge multicast filtering

Use the **bridge multicast filtering** Global Configuration mode command to enable the filtering of multicast addresses. Use the **no** form of this command to disable multicast address filtering.

#### Syntax

**bridge multicast filtering**

**no bridge multicast filtering**

#### Default Configuration

Multicast address filtering is disabled. All multicast addresses are flooded to all ports.

#### Command Mode

Global Configuration mode

#### User Guidelines

If multicast devices exist on the VLAN, do not change the unregistered multicast addresses' states to drop on the device ports.

If multicast devices exist on the VLAN and IGMP-snooping is not enabled, the **bridge multicast forward-all** command should be used to enable forwarding all multicast packets to the multicast switches.

#### Example

The following example enables bridge multicast filtering.

---

```
Console(config)# bridge multicast filtering
```

---

### 27.2 bridge multicast mode

Use the **bridge multicast mode** Interface Configuration (VLAN) mode command to configure the multicast bridging mode. Use the **no** form of this command to return to the default configuration.

**Syntax**

**bridge multicast mode** *{mac-group / ip-group / ip-src-group}*

**no bridge multicast mode**

**Parameters**

- **mac-group**—Specifies that multicast bridging is based on the packet's VLAN and MAC address.
- **ipv4-group**—Specifies that multicast bridging is based on the packet's VLAN and MAC address for non-IPv4 packets, and on the packet's VLAN and IPv4 destination address for IPv4 packets.
- **ipv4-src-group**—Specifies that multicast bridging is based on the packet's VLAN and MAC address for non-IPv4 packets, and on the packet's VLAN, IPv4 destination address and IPv4 source address for IPv4 packets.

**Default Configuration**

The default mode is mac-group.

**Command Mode**

Interface Configuration (VLAN) mode

**User Guidelines**

Use the mac-group mode when using a Network Management System that uses a MIB based on the multicast MAC address. Otherwise, it is recommended to use the ipv4-group or ipv4-src-group mode because there is no overlapping of IPv4 multicast addresses in these modes.

For each Forwarding Data Base (FDB) mode, use different CLI commands to configure static entries in the FDB, as described in the following table:

FDB mode	CLI commands	
<b>mac-group</b>	bridge multicast address	bridge multicast forbidden address
<b>ipv4-group</b>	bridge multicast ip-address	bridge multicast forbidden ip-address
<b>ipv4-src-group</b>	bridge multicast source group	bridge multicast forbidden source group

The following table describes the actual data that is written to the Forwarding Data Base (FDB) as a function of the IGMP version that is used in the network:

FDB mode	IGMP version 2	IGMP version 3
<b>mac-group</b>	MAC group address	MAC group address
<b>ipv4-group</b>	IP group address	IP group address
<b>ipv4-src-group</b>	(*)	IP source and group addresses

(\*) Note that (\*,G) cannot be written to the FDB if the mode is **ipv4-src-group**. In that case, no new FDB entry is created, but the port is added to the static (S,G) entries (if they exist) that belong to the requested group. It is recommended to set the FDB mode to **ipv4-group** or **mac-group** for IGMP version 2.

If an application on the device requests (\*,G), the operating FDB mode is changed to **ipv4-group**.

### Example

The following example configures the multicast bridging mode as **ipv4-group** on VLAN 2.

```
Console(config)# interface vlan 2
Console(config-if)# bridge multicast mode ipv4-group
```

## 27.3 bridge multicast address

Use the **bridge multicast address** Interface Configuration (VLAN) mode command to register a MAC-layer multicast address in the bridge table and statically add or remove ports to or from the group. Use the **no** form of this command to unregister the MAC address.

### Syntax

```
bridge multicast address {mac-multicast-address} [[add | remove] {ethernet interface-list | port-channel port-channel-list}]
```

```
no bridge multicast address {mac-multicast-address}
```

### Parameters

- **mac-multicast-address**—Specifies the group MAC multicast address.
- **add**—Adds ports to the group.
- **remove**—Removes ports from the group.
- **ethernet interface-list**—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- **port-channel port-channel-list**—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces; use a hyphen to designate a range of port channels.

### Default Configuration

No multicast addresses are defined.

If **ethernet interface-list** or **port-channel port-channel-list** is specified without specifying **add** or **remove**, the default option is **add**.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

To register the group in the bridge database without adding or removing ports or port channels, specify the **mac-multicast-address** parameter only.

Static multicast addresses can be defined on static VLANs only.

You can execute the command before the VLAN is created.

### Example

The following example registers the MAC address to the bridge table:

---

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast address 01:00:5e:02:02:03
```

---

The following example registers the MAC address and adds ports statically.

---

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast address 01:00:5e:02:02:03 add gil-2
```

---

## 27.4 bridge multicast forbidden address

Use the **bridge multicast forbidden address** Interface Configuration (VLAN) mode command to forbid adding or removing a specific multicast address to or from specific ports. Use the **no** form of this command to restore the default configuration.

### Syntax

**bridge multicast forbidden address** {*mac-multicast-address*} {*add* | *remove*}  
{*ethernet interface-list* | *port-channel port-channel-list*}

**no bridge multicast forbidden address** {*mac-multicast-address*}

### Parameters

- **mac-multicast-address**—Specifies the group MAC multicast address.
- **add**—Forbids adding ports to the group.
- **remove**—Forbids removing ports from the group.
- **ethernet *interface-list***—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- **port-channel *port-channel-list***—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces. Use a hyphen to designate a range of port channels.

### Default Configuration

No forbidden addresses are defined.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

Before defining forbidden ports, the multicast group should be registered.

You can execute the command before the VLAN is created.

### Example

The following example forbids MAC address 0100.5e02.0203 on port `gi9` within VLAN 8.

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast address 0100.5e02.0203
Console(config-if)# bridge multicast forbidden address 0100.5e02.0203 add gi9
```

## 27.5 bridge multicast ip-address

Use the **bridge multicast ip-address** Interface Configuration (VLAN) mode command to register IP-layer multicast addresses to the bridge table, and statically add or remove ports to or from the group. Use the `no` form of this command to unregister the IP address.

### Syntax

**bridge multicast ip-address** *ip-multicast-address* *[[add | remove] {ethernet interface-list | port-channel port-channel-list}*]

**no bridge multicast ip-address** *ip-multicast-address*

### Parameters

- **ip-multicast-address**—Specifies the group IP multicast address.
- **add**—Adds ports to the group.
- **remove**—Removes ports from the group.
- **ethernet interface-list**—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- **port-channel port-channel-list**—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces. Use a hyphen to designate a range of port channels.

### Default Configuration

No Multicast addresses are defined.

Default option is **add**.



## Command Mode

Interface Configuration (VLAN) mode

## User Guidelines

To register the group in the bridge database without adding or removing ports or port channels, specify the **ip-multicast-address** parameter only.

Static multicast addresses can be defined on static VLANs only.

You can execute the command before the VLAN is created.

## Example

The following example registers the IP address to the bridge table:

---

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast ip-address 239.2.2.2
```

---

The following example registers the IP address and adds ports statically.

---

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast ip-address 239.2.2.2 add gi9
```

---

## 27.6 bridge multicast forbidden ip-address

Use the **bridge multicast forbidden ip-address** Interface Configuration (VLAN) mode command to forbid adding or removing a specific IP multicast address to or from specific ports. Use the no form of this command to restore the default configuration.

### Syntax

```
bridge multicast forbidden ip-address {ip-multicast-address} {add | remove}
{ethernet interface-list | port-channel port-channel-list}
```

```
no bridge multicast forbidden ip-address {ip-multicast-address}
```

### Parameters

- **ip-multicast-address**—Specifies the group IP multicast address.

- **add**—Forbids adding ports to the group.
- **remove**—Forbids removing ports from the group.
- **ethernet *interface-list***—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- **port-channel *port-channel-list***—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces. Use a hyphen to designate a range of port channels.

### Default Configuration

No forbidden addresses are defined.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

Before defining forbidden ports, the multicast group should be registered.

You can execute the command before the VLAN is created.

### Example

The following example registers IP address 239.2.2.2, and forbids the IP address on port `gi9` within VLAN 8.

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast ip-address 239.2.2.2
Console(config-if)# bridge multicast forbidden ip-address 239.2.2.2 add gi9
```

## 27.7 bridge multicast source group

Use the **bridge multicast source group** Interface Configuration (VLAN) mode command to register a source IP address - multicast IP address pair to the bridge table, and statically add or remove ports to or from the source-group. Use the `no` form of this command to unregister the source-group-pair.

### Syntax

**bridge multicast source** *ip-address* **group** *ip-multicast-address* *[[add | remove]*  
*{ethernet interface-list | port-channel port-channel-list}*

**no bridge multicast source** *ip-address* **group** *ip-multicast-address*

### Parameters

- **ip-address**—Specifies the source IP address.
- **ip-multicast-address**—Specifies the group IP multicast address.
- **add**—Adds ports to the group for the specific source IP address.
- **remove**—Removes ports from the group for the specific source IP address.
- **ethernet** *interface-list*—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- **port-channel** *port-channel-list*—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces; use a hyphen to designate a range of port channels.

### Default Configuration

No multicast addresses are defined.

The default option is **add**.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

You can execute the command before the VLAN is created.

### Example

The following example registers a source IP address - multicast IP address pair to the bridge table:

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast source 239.2.2.2 group 239.2.2.2
```

## 27.8 bridge multicast forbidden source group

Use the **bridge multicast forbidden source group** Interface Configuration (VLAN) mode command to forbid adding or removing a specific IP source address - multicast address pair to or from specific ports. Use the no form of this command to return to the default configuration.

### Syntax

```
bridge multicast forbidden source ip-address group ip-multicast-address {add /  
remove} {ethernet interface-list | port-channel port-channel-list}
```

```
no bridge multicast forbidden source ip-address group ip-multicast-address
```

### Parameters

- **ip-address**—Specifies the source IP address.
- **ip-multicast-address**—Specifies the group IP multicast address.
- **add**—Forbids adding ports to the group for the specific source IP address.
- **remove**—Forbids removing ports from the group for the specific source IP address.
- **ethernet** *interface-list*—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- **port-channel** *port-channel-list*—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces; use a hyphen to designate a range of port channels.

### Default Configuration

No forbidden addresses are defined.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

Before defining forbidden ports, the multicast group should be registered.

You can execute the command before the VLAN is created.

### Example

The following example registers a source IP address - multicast IP address pair to the bridge table, and forbids adding the pair to gigabitethernet port `gi9` on VLAN 8:

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast source 13.16.1.1 group 239.2.2.2
Console(config-if)# bridge multicast forbidden source 13.16.1.1 group
239.2.2.2 add gi9
```

## 27.9 bridge multicast ipv6 mode

Use the **bridge multicast ipv6 mode** Interface Configuration (VLAN) mode command to configure the multicast bridging mode for ipv6 multicast packets. Use the no form of this command to return to the default configuration.

### Syntax

```
bridge multicast ipv6 mode {mac-group / ip-group / ip-src-group}
```

```
no bridge multicast ipv6 mode
```

### Parameters

- **mac-group**—Specifies that multicast bridging is based on the packet's VLAN and MAC address.
- **ip-group**—Specifies that multicast bridging is based on the packet's VLAN and IPv6 destination address for IPv6 packets.
- **ip-src-group**—Specifies that multicast bridging is based on the packet's VLAN, IPv6 destination address and IPv6 source address for IPv6 packets.

### Default Configuration

The default mode is `mac-group`.

### Command Mode

Interface Configuration (VLAN) mode

## User Guidelines

Use the **mac-group** mode when using a Network Management System that uses a MIB based on the multicast MAC address.

For each Forwarding Data Base (FDB) mode, use different CLI commands to configure static entries for IPv6 multicast addresses in the FDB, as described in the following table:

FDB mode	CLI commands	
<b>mac-group</b>	bridge multicast address	bridge multicast forbidden address
<b>ipv4-group</b>	bridge multicast ipv6 ip-address	bridge multicast ipv6 forbidden ip-address
<b>ipv4-src-group</b>	bridge multicast ipv6 source group	bridge multicast ipv6 forbidden source group

The following table describes the actual data that is written to the Forwarding Data Base (FDB) as a function of the MLD version that is used in the network:

FDB mode	MLD version 1	MLD version 2
<b>mac-group</b>	MAC group address	MAC group address
<b>ipv4-group</b>	IP group address	IP group address
<b>ipv4-src-group</b>	(*)	IP source and group addresses

(\*) Note that (\*,G) cannot be written to the FDB if the mode is **ip-src-group**. In that case, no new FDB entry is created, but the port is added to the (S,G) entries (if they exist) that belong to the requested group. If an application on the device requests (\*,G), the operating FDB mode is changed to **ip-group**.

You can execute the command before the VLAN is created.

## Example

The following example configures the Multicast bridging mode as **ip-group** on VLAN 2.

```
Console(config)# interface vlan 2
Console(config-if)# bridge multicast ipv6 mode ip-group
```

## 27.10 bridge multicast ipv6 ip-address

Use the **bridge multicast ipv6 ip-address** Interface Configuration (VLAN) mode command to register an IPv6 multicast address to the bridge table, and statically add or remove ports to or from the group. Use the **no** form of this command to unregister the IPv6 address.

### Syntax

```
bridge multicast ipv6 ip-address ipv6-multicast-address [[add | remove] {ethernet interface-list | port-channel port-channel-list}]
```

```
no bridge multicast ipv6 ip-address ip-multicast-address
```

### Parameters

- **ipv6-multicast-address**—Specifies the group IPv6 multicast address.
- **add**—Adds ports to the group.
- **remove**—Removes ports from the group.
- **ethernet interface-list**—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces; use a hyphen to designate a range of ports.
- **port-channel port-channel-list**—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces. Use a hyphen to designate a range of port channels.

### Default Configuration

No multicast addresses are defined.

The default option is **add**.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

To register the group in the bridge database without adding or removing ports or port channels, specify the **ipv6-multicast-address** parameter only.

Static Multicast addresses can be defined on static VLANs only.

You can execute the command before the VLAN is created.

### Example

The following example registers the IPv6 address to the bridge table:

---

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast ipv6 ip-address FF00:0:0:0:4:4:4
```

---

The following example registers the IPv6 address and adds ports statically.

---

```
console(config)# interface vlan 8
Console(config-if)# bridge multicast ipv6 ip-address FF00:0:0:0:4:4:4 add
gil-2
```

---

## 27.11 bridge multicast ipv6 forbidden ip-address

Use the **bridge multicast ipv6 forbidden ip-address** Interface Configuration (VLAN) mode command to forbid adding or removing a specific IPv6 multicast address to or from specific ports. To restore the default configuration, use the **no** form of this command.

### Syntax

```
bridge multicast ipv6 forbidden ip-address {ipv6-multicast-address} {add / remove} {ethernet interface-list / port-channel port-channel-list}
```

```
no bridge multicast ipv6 forbidden ip-address {ipv6-multicast-address}
```

### Parameters

- **ipv6-multicast-address**—Specifies the group IPv6 multicast address.
- **add**—Forbids adding ports to the group.
- **remove**—Forbids removing ports from the group.
- **ethernet interface-list**—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.



- **port-channel** *port-channel-list*—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces. Use a hyphen to designate a range of port channels.

### Default Configuration

No forbidden addresses are defined.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

Before defining forbidden ports, the multicast group should be registered.

You can execute the command before the VLAN is created.

### Example

The following example registers an IPv6 multicast address, and forbids the IPv6 address on port `gi9` within VLAN 8.

```
console(config)# interface vlan 8
Console(config-if)# bridge multicast ipv6 ip-address FF00:0:0:0:4:4:4
Console(config-if)# bridge multicast ipv6 forbidden ip-address
FF00:0:0:0:4:4:4 add gi9
```

## 27.12 bridge multicast ipv6 source group

Use the **bridge multicast ipv6 source group** Interface Configuration (VLAN) mode command to register a source IPv6 address - multicast IPv6 address pair to the bridge table, and statically add or remove ports to or from the source-group. Use the **no** form of this command to unregister the source-group-pair.

### Syntax

```
bridge multicast ipv6 source ipv6-source-address group ipv6-multicast-address
[[add | remove] [ethernet interface-list | port-channel port-channel-list]]
```

```
no bridge multicast ipv6 source ipv6-address group ipv6-multicast-address
```

### Parameters

- **ipv6-source-address**—Specifies the source IPv6 address.
- **ipv6-multicast-address**—Specifies the group IPv6 multicast address.
- **add**—Adds ports to the group for the specific source IPv6 address.
- **remove**—Removes ports from the group for the specific source IPv6 address.
- **ethernet *interface-list***—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- **port-channel *port-channel-list***—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces. Use a hyphen to designate a range of port channels.

### Default Configuration

No multicast addresses are defined.

If **ethernet *interface-list*** or **port-channel *port-channel-list*** is specified without specifying **add** or **remove**, the default option is **add**.

You can execute the command before the VLAN is created.

### Command Mode

Interface Configuration (VLAN) mode

### Example

The following example registers a source IPv6 address - multicast IPv6 address pair to the bridge table:

```
Console(config)# interface vlan 8
Console(config-if)# bridge multicast source 2001:0:0:0:4:4:4 group
FF00:0:0:0:4:4:4
```

---

## 27.13 bridge multicast ipv6 forbidden source group

Use the **bridge multicast ipv6 forbidden source group** Interface Configuration (VLAN) mode command to forbid adding or removing a specific IP source address

- multicast address pair to or from specific ports. Use the **no** form of this command to return to the default configuration.

### Syntax

**bridge multicast ipv6 forbidden source** *ipv6-source-address* **group** *ipv6-multicast-address* {**add** | **remove**} {**ethernet** *interface-list* | **port-channel** *port-channel-list*}

**no bridge multicast ipv6 forbidden source** *ipv6-address* **group** *ipv6-multicast-address*

### Parameters

- **ipv6-source-address**—Specifies the source IPv6 address.
- **ipv6-multicast-address**—Specifies the group IPv6 multicast address.
- **add**—Forbids adding ports to the group for the specific source IPv6 address.
- **remove**—Forbids removing ports from the group for the specific source IPv6 address.
- **ethernet** *interface-list*—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- **port-channel** *port-channel-list*—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces; use a hyphen to designate a range of port channels.

### Default Configuration

No forbidden addresses are defined.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

Before defining forbidden ports, the multicast group should be registered.

You can execute the command before the VLAN is created.

### Example

The following example registers a source IPv6 address - multicast IPv6 address pair to the bridge table, and forbids adding the pair to `gi9` on VLAN 8:

```
Console(config)# interface vlan 8

Console(config-if)# bridge multicast source 2001:0:0:0:4:4:4 group
FF00:0:0:0:4:4:4

Console(config-if)# bridge multicast forbidden source 2001:0:0:0:4:4:4 group
FF00:0:0:0:4:4:4 add gi9
```

## 27.14 bridge multicast unregistered

Use the **bridge multicast unregistered** Interface Configuration (Ethernet, Port-Channel) mode command to configure the forwarding state of unregistered multicast addresses. Use the **no** form of this command to restore the default configuration.

### Syntax

**bridge multicast unregistered** *{forwarding / filtering}*

**no bridge multicast unregistered**

### Parameters

- **forwarding**—Forwards unregistered multicast packets.
- **filtering**—Filters unregistered multicast packets.

### Default Configuration

Unregistered multicast addresses are forwarded.

### Command Mode

Interface Configuration (Ethernet, Port-Channel) mode

### User Guidelines

Do not enable unregistered multicast filtering on ports that are connected to routers, because the 224.0.0.x address range should not be filtered. Note that routers do not necessarily send IGMP reports for the 224.0.0.x range.

You can execute the command before the VLAN is created.

### Example

The following example specifies that unregistered multicast packets are filtered on `gi1`:

```
Console(config)# interface gi1
Console(config-if)# bridge multicast unregistered filtering
```

---

## 27.15 bridge multicast forward-all

Use the **bridge multicast forward-all** Interface Configuration (VLAN) mode command to enable forwarding all multicast packets for a range of ports or port channels. Use the **no** form of this command to restore the default configuration.

### Syntax

**bridge multicast forward-all** *{add / remove}* *{ethernet interface-list / port-channel port-channel-list}*

**no bridge multicast forward-all**

### Parameters

- **add**—Forces forwarding of all multicast packets.
- **remove**—Does not force forwarding of all multicast packets.
- **ethernet interface-list**—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- **port-channel port-channel-list**—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces. Use a hyphen to designate a range of port channels.

### Default Configuration

Forwarding of all multicast packets is disabled.

### Command Mode

Interface Configuration (VLAN) mode

## Example

The following example enables all multicast packets on port gi8 to be forwarded.

```
Console(config)# interface vlan 2
Console(config-if)# bridge multicast forward-all add gi8
```

## 27.16 bridge multicast forbidden forward-all

Use the **bridge multicast forbidden forward-all** Interface Configuration (VLAN) mode command to forbid a port to dynamically join multicast groups. Use the **no** form of this command to restore the default configuration.

### Syntax

```
bridge multicast forbidden forward-all {add | remove} {ethernet interface-list /  
port-channel port-channel-list}
```

```
no bridge multicast forbidden forward-all
```

### Parameters

- **add**—Forbids forwarding of all multicast packets.
- **remove**—Does not forbid forwarding of all multicast packets.
- ***ethernet interface-list***—Specifies a list of Ethernet ports. Separate nonconsecutive Ethernet ports with a comma and no spaces. Use a hyphen to designate a range of ports.
- ***port-channel port-channel-list***—Specifies a list of port channels. Separate nonconsecutive port-channels with a comma and no spaces; use a hyphen to designate a range of port channels.

### Default Configuration

Ports are not forbidden to dynamically join multicast groups.

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

Use this command to forbid a port to dynamically join (by IGMP, for example) a multicast group.

The port can still be a multicast router port.

### Example

The following example forbids forwarding of all multicast packets to gi1 within VLAN 2.

```
Console(config)# interface vlan 2
Console(config-if)# bridge multicast forbidden forward-all add ethernet gi1
```

## 27.17 mac address-table static

Use the **mac address-table static** Global Configuration mode command to add MAC-layer station source address to the MAC address table. Use the **no** form of this command to delete the MAC address.

### Syntax

```
mac address-table static mac-address vlan vlan-id interface interface-id
[permanent / delete-on-reset / delete-on-timeout / secure]
```

```
no mac address-table static [mac-address] vlan vlan-id
```

### Parameters

**mac-address**—AC address (Range: Valid MAC address)

**vlan-id**—Specify the VLAN

**interface-id**—Specify an interface ID. The interface ID can be one of the following types: Ethernet port or port-channel (Range: Valid Ethernet port, Valid Port-channel number)

**permanent**—The address can only be deleted by the **no bridge address** command.

**delete-on-reset**—The address is deleted after reset.

**delete-on-timeout**—The address is deleted after aged out.

**secure**—The address is deleted after the port changes mode to unlock learning (no port security command). Available only when the port is in learning locked mode.

### Default Configuration

No static addresses are defined. The default mode for an added address is permanent.

### Command Mode

Global Configuration mode

### Example

```
console(config)# mac address-table static 00:3f:bd:45:5a:b1 vlan 1 gil
```

---

## 27.18 clear mac address-table

Use the **clear mac address-table** Privileged EXEC command to remove learned or secure entries from the forwarding database.

### Syntax

**clear mac address-table** *dynamic* [*interface interface-id*]

**clear mac address-table** *secure interface interface-id*

### Parameters

**interface** *interface-id*—Delete all dynamic address on the specified interface. The interface ID can be one of the following types: Ethernet port or port-channel.

### Command Mode

Privileged EXEC mode

### Example

```
console# clear mac address-table dynamic
```

---

## 27.19 mac address-table aging-time

Use the **mac address-table aging-time** global configuration command to set the aging time of the address table. Use the **no** form of this command to restore the default.



**Syntax**

**mac address-table aging-time** *seconds*

**no mac address-table aging-time**

**Parameters**

**seconds**—Time is number of seconds. (Range:10–630)

**Default Configuration**

300

**Command Mode**

Global Configuration mode

**Example**

---

```
console(config)# mac address-table aging-time 600
```

---

**27.20 port security**

Use the **port security** Interface Configuration (Ethernet, Port-channel) mode command to enable port security on an interface. Use the **no** form of this command to disable port security on an interface.

**Syntax**

**port security** [*forward* / *discard* / *discard-shutdown*] [*trap seconds*]

**no port security**

**Parameters**

- **forward**—Forwards packets with unlearned source addresses, but does not learn the address.
- **discard**—Discards packets with unlearned source addresses.
- **discard-shutdown**—Discards packets with unlearned source addresses and shuts down the port.
- **trap** *seconds*—Sends SNMP traps and specifies the minimum time interval in seconds between consecutive traps. (Range: 1–1000000)

### Default Configuration

The feature is disabled

The default mode is discard.

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### Example

The following example forwards all packets to port gi 1 without learning addresses of packets from unknown sources and sends traps every 100 seconds if a packet with an unknown source address is received.

---

```
console(config)# gi1
Console(config-if)# port security forward trap 100
```

---

## 27.21 port security mode

Use the **port security mode** Interface Configuration (Ethernet, port-channel) mode command configures the port security learning mode. Use the no form of this command to restore the default configuration.

### Syntax

**port security mode** {lock | max-addresses}

**no port security mode**

### Parameters

- **lock**—Saves the current dynamic MAC addresses associated with the port and disables learning, relearning and aging.
- **max-addresses**—Deletes the current dynamic MAC addresses associated with the port and learns up to the maximum number of addresses allowed on the port. Relearning and aging are enabled.

### Default Configuration

The default port security mode is lock.

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### Example

The following example sets the port security mode to Lock for gi7.

---

```
Console(config)# interface gi7
Console(config-if)# port security mode lock
```

---

## 27.22 port security max

Use the **port security mode** Interface Configuration (Ethernet, Port-channel) mode command to configure the maximum number of addresses that can be learned on the port while the port is in port security max-addresses mode. Use the **no** form of this command to restore the default configuration.

### Syntax

**port security max** {*max-addr*}

**no port security max**

### Parameters

**max-addr**—Specifies the maximum number of addresses that can be learned on the port. (Range: 0–256)

### Default Configuration

This default maximum number of addresses is 1.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

This command is relevant in port security max-addresses mode only.

### Example

---

```
Console(config)# interface gi1
```

---

```
Console(config-if)# port security max 20
```

---

## 27.23 show mac address-table

Use the **show mac address-table** EXEC command to view entries in the MAC address table.

### Syntax

**show mac address-table** [*dynamic* / *static* / *secure*] [*vlan* *vlan*] [*interface* *interface-id*] [*address* *mac-address*]

### Parameters

- **dynamic**—Displays only dynamic MAC address table entries.
- **static**—Displays only static MAC address table entries.
- **secure**—Displays only secure MAC address table entries.
- **vlan**—Specifies VLAN, such as VLAN 1.
- **interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or port-channel.
- **mac-address**—MAC address.

### Default Configuration

If no parameters are entered, the entire table is displayed.

### Command Mode

EXEC mode

### User Guidelines

Internal usage VLANs (VLANs that are automatically allocated on routed ports) are presented in the VLAN column by a port number and not by a VLAN ID.

## Example

```
Console# show mac address-table
```

```
Aging time is 300 sec
```

VLAN	MAC Address	Port	Type
1	00:00:26:08:13:23	0	self
1	00:3f:bd:45:5a:b1	gi1	static
1	00:a1:b0:69:63:f3	gi4	dynamic
2	00:a1:b0:69:63:f3	gi5	dynamic

```
Console# show mac address-table 00:3f:bd:45:5a:b1
```

```
Aging time is 300 sec
```

VLAN	MAC Address	Port	Type
1	00:3f:bd:45:5a:b1	static	gi9

## 27.24 show mac address-table count

Use the **show mac address-table count** EXEC mode command to display the number of addresses present in the Forwarding Database.

### Syntax

```
show mac address-table count [vlan vlan / interface interface-id]
```

### Parameters

- **vlan**—Specifies VLAN.
- **interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or port-channel.

## Command Mode

EXEC mode

## Example

---

```
Console# show mac address-table count
```

```
Capacity: 8192
```

```
Free: 8083
```

```
Used: 109
```

```
Static addresses: 2
```

```
Secure addresses: 1
```

```
Dynamic addresses: 97
```

```
Internal addresses: 9
```

---

## 27.25 show bridge multicast mode

Use the **show bridge multicast mode** EXEC mode command to display the multicast bridging mode for all VLANs or for a specific VLAN.

### Syntax

```
show bridge multicast mode [vlan vlan-id]
```

### Parameters

**vlan *vlan-id***—Specifies the VLAN ID.

### Command Mode

EXEC mode

## Example

The following example displays the multicast bridging mode for all VLANs.

```
Console# show bridge multicast mode
```

VLAN	IPv4 Multicast mode		IPv6 Multicast mode	
	Admin	Oper	Admin	Oper
---	-----	-----	-----	-----
1	MAC-GROUP	MAC-GROUP	MAC-GROUP	-MAC-GROUP
11	IPv4-GROUP	IPv4-GROUP	IPv6-GROUP	IPv6-GROUP
12	IPv4-SRC- GROUP	IPv4-SRC- GROUP	IPv6-SRC- GROUP	IPv6-SRC- GROUP

## 27.26 show bridge multicast address-table

Use the **show bridge multicast address-table** EXEC mode command to display multicast MAC address or IP address table information.

### Syntax

```
show bridge multicast address-table [vlan vlan-id] [address
{mac-multicast-address | ipv4-multicast-address | ipv6-multicast-address}] [format
{ip | mac}] [source {ipv4-source-address | ipv6-source-address}]
```

### Parameters

- **vlan-id**—Specifies the VLAN ID.
- **address {mac-multicast-address | ipv4-multicast-address | ipv6-multicast-address}**—Specifies the multicast address. The possible values are:
  - **mac-multicast-address**—Specifies the MAC multicast address.
  - **ipv4-multicast-address**—Specifies the IPv4 multicast address.
  - **ipv6-multicast-address**—Specifies the IPv6 multicast address.
- **format {ip | mac}**—Specifies the multicast address format. The possible values are:
  - **ip**—Specifies that the multicast address is an IP address.

- **mac**—Specifies that the multicast address is a MAC address.
- **source {ipv4-source-address | ipv6-source-address}**—Specifies the source address. The possible values are:
  - **ipv4-address**—Specifies the source IPv4 address.
  - **ipv6-address**—Specifies the source IPv6 address.

### Default Configuration

If the format is not specified, it defaults to **mac**.

### Command Mode

EXEC mode

### User Guidelines

A MAC address can be displayed in IP format only if it is within the range 0100.5e00.0000 through 0100.5e7f.ffff.

Multicast Router ports (defined statically or discovered dynamically) are members in all MC groups.

Ports that were defined via **bridge multicast forbidden forward-all** command are displayed in all forbidden MC entries.

Changing the multicast mode can move static multicast addresses that are written in the device FDB to a shadow configuration because of FDB hash collisions.

### Example

The following example displays bridge multicast address information.

```

Console# show bridge multicast address-table

Multicast address table for VLANs in MAC-GROUP bridging mode:

Vlan    MAC Address          Type          Ports
----  -
8       01:00:5e:02:02:03   Static        1-2

Forbidden ports for multicast addresses:

Vlan    MAC Address          Ports
----  -

```



```
8 01:00:5e:02:02:03 gi9
```

Multicast address table for VLANs in IPv4-GROUP bridging mode:

Vlan	MAC Address	Type	Ports
1	224.0.0.251	Dynamic	gi2

Forbidden ports for multicast addresses:

Vlan	MAC Address	Ports
1	232.5.6.5	
1	233.22.2.6	

Multicast address table for VLANs in IPv4-SRC-GROUP bridging mode:

Vlan	Group Address	Source address	Type	Ports
1	224.2.2.251	11.2.2.3	Dynamic	gi1

Forbidden ports for multicast addresses:

Vlan	Group Address	Source Address	Ports
8	239.2.2.2	*	gi9
8	239.2.2.2	1.1.1.11	gi9

Multicast address table for VLANs in IPv6-GROUP bridging mode:

VLAN	IP/MAC Address	Type	Ports
8	ff02::4:4:4	Static	gi1-2, gi7, Po1

Forbidden ports for multicast addresses:

VLAN	IP/MAC Address	Ports
8	ff02::4:4:4	gi9

Multicast address table for VLANs in IPv6-SRC-GROUP bridging mode:

Vlan	Group	Address	Source address	Type	Ports
8	ff02::4:4:4	*		Static	gi1-2,gi7,Pol
8	ff02::4:4:4		fe80::200:7ff: fe00:200	Static	

Forbidden ports for multicast addresses:

Vlan	Group	Address	Source address	Ports
8	ff02::4:4:4	*		gi9
8	ff02::4:4:4		fe80::200:7ff:f e00:200	gi9

## 27.27 show bridge multicast unregistered

Use the **show bridge multicast unregistered** EXEC mode command to display the unregistered multicast filtering configuration.

### Syntax

**show bridge multicast unregistered** *[interface-id]*

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

### Command Mode

EXEC mode

**Example**

The following example displays the unregistered multicast configuration.

---

```
Console# show bridge multicast unregistered
```

```
Port          Unregistered
```

```
-----
```

```
gi1          Forward
```

```
gi2          Filter
```

```
gi3          Filter
```

**27.28 show ports security**

Use the **show ports security** Privileged EXEC mode command to display the port-lock status.

**Syntax**

```
show ports security [interface-id]
```

**Parameters**

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays the port-lock status of all ports.

---

```
console# show ports security
```

```
Port   Status   Learning   Action   Max   Trap   Frequency
```

```
-----
```

```
gi1    Enabled  Max-      Discard  3     Enabled 100
```

```
Addresses
```

```

gi2    Disabled Max-      -      28    -      -
        Addresses
gi3    Enabled Lock      Discard, 8    Disabled -
        Shutdown

```

The following table describes the fields shown above.

Field	Description
<b>Port</b>	The port number.
<b>Status</b>	The port security status. The possible values are: Enabled or Disabled.
<b>Mode</b>	The port security mode.
<b>Action</b>	The action taken on violation.
<b>Maximum</b>	The maximum number of addresses that can be associated on this port in the Max-Addresses mode.
<b>Trap</b>	The status of SNMP traps. The possible values are: Enable or Disable.
<b>Frequency</b>	The minimum time interval between consecutive traps.

## 27.29 show ports security addresses

Use the **show ports security addresses** Privileged EXEC mode command to display the current dynamic addresses in locked ports.

### Syntax

```
show ports security addresses [interface-id]
```

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

### Command Mode

Privileged EXEC mode

### Example

The following example displays dynamic addresses in all currently locked ports.

```
Console# show ports security addresses
```

Port	Status	Learning	Current	Maximum
----	-----	-----	-----	-----
gi1	Enabled	Max-addresses	2	3
gi2	Disabled	Max-addresses	-	128
gi3	Enabled	Lock	NA	NA

## 27.30 bridge multicast reserved-address

Use the **bridge multicast reserved-address** Global Configuration mode command to define the action on multicast reserved-address packets. Use the **no** form of this command to revert to default.

### Syntax

```
bridge multicast reserved-address mac-multicast-address [ethernet-v2 ethtype / llc sap / llc-snap pid] {discard | bridge}
```

```
no bridge multicast reserved-address mac-multicast-address [ethernet-v2 ethtype / llc sap / llc-snap pid]
```

### Parameters

- **mac-multicast-address**—MAC multicast address in the reserved MAC addresses range.(Range: 01-80-C2-00-00-00, 01-80-C2-00-00-02–01-80-C2-00-00-2F)
- **ethernet-v2 ethtype**—Specifies that the packet type is Ethernet v2 and the Ethernet type field (16 bits in hexadecimal format).(Range: 0x0600–0xFFFF)
- **llc sap**—Specifies that the packet type is LLC and the DSAP-SSAP field (16 bits in hexadecimal format).(Range: 0xFFFF)
- **llc-snap pid**—Specifies that the packet type is LLC-SNAP and the PID field (40 bits in hexadecimal format). (Range: 0x0000000000 - 0xFFFFFFFF)
- **discard**—Specifies discarding the packets.
- **bridge**—Specifies bridging the packets

### Default Configuration

If an **<address, frame type, protocol>** tuple is mapped to an application that is supported by the device then the default is Peer (handled by the application rules). Otherwise:

For addresses in the range 01-80-C2-00-00-00, 01-80-C2-00-00-02–01-80-C2-00-00-0F, the default is discard. For addresses in the range 00-80-C2-00-00-10– 01-80-C2-00-00-2F, the default is bridge.

### Command Mode

Global Configuration mode

### User Guidelines

If the packet/service type is not specified, the configuration is relevant to all the packets with the configured MAC address.

Specific configurations (that contain service type) have precedence over less specific configurations (contain only MAC address).

The packets that are bridged are subject to security ACLs.

The actions define by this command has precedence over forwarding rules defined by applications/protocols (STP, LLDP etc.) supported on the device.

---

## 27.31 show bridge multicast reserved-addresses

Use the **show bridge multicast reserved-addresses** EXEC mode command to display the multicast reserved-address rules.

### Syntax

**show bridge multicast reserved-addresses**

### Command Mode

EXEC mode

### Example

```
Console # show bridge multicast reserved-addresses
```

```
MAC addressFrame typeProtocolAction
```

```
-----
```

---

01-80-C2-00-00-00LLC-SNAP00-00-0C-01-29Bridge

## 28 Port Monitor Commands

### 28.1 port monitor

Use the **port monitor** Interface Configuration (Ethernet) mode command to start a port monitoring session (mirroring). Use the **no** form of this command to stop a port monitoring session.

#### Syntax

```
port monitor src-interface-id [rx | tx]
```

```
no port monitor src-interface-id
```

```
port monitor vlan vlan-id
```

```
no port monitor vlan vlan-id
```

#### Parameters

- **rx**—Monitors received packets only. If no option is specified, it monitors both rx and tx.
- **tx**—Monitors transmitted packets only. If no option is specified, it monitors both rx and tx.
- **vlan** *vlan-id*—VLAN number
- **src-interface-id**—Specifies an interface ID. The interface ID must be an Ethernet port.

#### Default Configuration

Monitors both received and transmitted packets.

#### Command Mode

Interface Configuration (Ethernet) mode. It cannot be configured for a range of interfaces (range context).

#### User Guidelines

This command enables port copy between Source Port (*src-interface*) to a Destination Port (The port in context).

The analyzer port for port ingress traffic mirroring should be the same port for all mirrored ports.



The analyzer port for port egress traffic mirroring should be the same port for all mirrored ports.

The analyzer port for VLAN mirroring should be the same for all the mirrored VLANs, and should be the same port as the analyzer port for port ingress mirroring traffic.

Following are restrictions apply for ports that are configured to be source ports:

- The port cannot be a destination port.

The following restrictions apply to ports that are configured to be monitor ports:

- The port can't be source port.
- The port isn't member in port-channel.
- IP interface is not configured on the port.
- GVRP is not enabled on the port.
- The port is not a member in any VLAN, except for the default VLAN (will be automatically removed from the default VLAN).
- L2 protocols, such as: LLDP, CDP, LBD, STP, LACP, are not active on the destination port.

The following restrictions apply to ports that are configured to be monitor ports:

- The port cannot be source port.
- The port is not a member in port-channel.

Notes:

1. In this mode some traffic duplication on the analyzer port may be observed. For example:
  - Port 2 is being egress monitored by port 4.
  - Port 2 & 4 are members in VLAN 3.
  - Unknown Unicast packet sent to VLAN 3 will egress from port 4 twice, one instance as normal forward and another instance as mirrored from port 2.
  - Moreover, if port 2 is an untagged member in VLAN 3 and port 4 is a tagged member then both instances will look different (one tagged and the other is not).
2. When the port is configured to 802.1X auto mode it will forward any mirrored traffic regardless of the 1X state. However, it will operate as a normal network port (forward traffic) only after authorization is done.

- Mirrored traffic is exposed to STP state, i.e. if the port is in STP blocking, it will not egress any mirrored traffic.

### Example

The following example copies traffic for both directions (Tx and Rx) from the source port `gi1/1/2` to destination port `gi1/1/1`.

---

```
Console(config)# interface gi1/1/1
Console(config-if)# port monitor gi1/1/2
```

---

## 28.2 show ports monitor

Use the **show ports monitor** EXEC mode command to display the port monitoring status.

### Syntax

**show ports monitor**

### Command Mode

EXEC mode

### Example

The following example displays the port monitoring status.

---

```
Console# show ports monitor
```

Source port	Destination Port	Type	Status
-----	-----	-----	-----
gi1/1/8	gi1/1/1		RX,TX Active
gi1/1/2	gi1/1/1		RX,TX Active
gi1/1/18	gi1/1/1	Rx	Active
VLAN 9	gi1/1/1	N/A	Active

---

## 29 Spanning-Tree Commands

---

### 29.1 spanning-tree

Use the **spanning-tree** Global Configuration mode command to enable spanning-tree functionality. Use the **no** form of this command to disable the spanning-tree functionality.

#### Syntax

**spanning-tree**

**no spanning-tree**

#### Default Configuration

Spanning-tree is enabled.

#### Command Mode

Global Configuration mode

#### Example

The following example enables spanning-tree functionality.

---

```
Console(config)# spanning-tree
```

---

### 29.2 spanning-tree mode

Use the **spanning-tree mode** Global Configuration mode command to configure the spanning-tree protocol currently running. Use the **no** form of this command to restore the default configuration.

#### Syntax

**spanning-tree mode** *{stp / rstp / mst}*

**no spanning-tree mode**

#### Parameters

- **stp**—Specifies that the Spanning Tree Protocol (STP) is enabled.

- **rstp**—Specifies that the Rapid Spanning Tree Protocol (RSTP) is enabled.
- **mst**—Specifies that the Multiple Spanning Tree Protocol (MSTP) is enabled.

### Default Configuration

The default is RSTP.

### Command Mode

Global Configuration mode

### User Guidelines

In RSTP mode, the device uses STP when the neighbor device uses STP.

In MSTP mode, the device uses RSTP when the neighbor device uses RSTP, and uses STP when the neighbor device uses STP.

### Example

The following example configures the spanning-tree protocol as MSTP.

---

```
console(config)# spanning-tree mode mstp
```

---

## 29.3 spanning-tree forward-time

Use the **spanning-tree forward-time** Global Configuration mode command to configure the spanning-tree bridge forward time, which is the amount of time a port remains in the listening and learning states before entering the forwarding state. Use the **no** form of this command to restore the default configuration.

### Syntax

**spanning-tree forward-time** *seconds*

**no spanning-tree forward-time**

### Parameters

**seconds**—Specifies the spanning-tree forward time in seconds. (Range: 4–30)

### Default Configuration

The default forwarding time for the IEEE Spanning Tree Protocol (STP) is 15 seconds.

### Command Mode

Global Configuration mode

### User Guidelines

When configuring the forwarding time, the following relationship should be maintained:

$$2 * (\text{Forward-Time} - 1) \geq \text{Max-Age}$$

### Example

The following example configures the spanning tree bridge forwarding time to 25 seconds.

---

```
Console(config)# spanning-tree forward-time 25
```

---

## 29.4 spanning-tree hello-time

Use the **spanning-tree hello-time** Global Configuration mode command to configure the spanning tree bridge Hello time, which is how often the device broadcasts Hello messages to other devices. Use the **no** form of this command to restore the default configuration.

### Syntax

**spanning-tree hello-time** *seconds*

**no spanning-tree hello-time**

### Parameters

**seconds**—Specifies the spanning-tree Hello time in seconds. (Range: 1–10)

### Default Configuration

The default Hello time for IEEE Spanning Tree Protocol (STP) is 2 seconds.

### Command Mode

Global Configuration mode

### User Guidelines

When configuring the Hello time, the following relationship should be maintained:

$$\text{Max-Age} \geq 2 * (\text{Hello-Time} + 1)$$

### Example

The following example configures the spanning-tree bridge hello time to 5 seconds.

---

```
Console(config)# spanning-tree hello-time 5
```

---

## 29.5 spanning-tree max-age

Use the **spanning-tree max-age** Global Configuration mode command to configure the spanning-tree bridge maximum age. Use the **no** form of this command to restore the default configuration.

### Syntax

**spanning-tree max-age** *seconds*

**no spanning-tree max-age**

### Parameters

**seconds**—Specifies the spanning-tree bridge maximum age in seconds. (Range: 6–40)

### Default Configuration

The default maximum age for IEEE Spanning Tree Protocol (STP) is 20 seconds.

### Command Mode

Global Configuration mode

### User Guidelines

When configuring the maximum age, the following relationships should be maintained:

$$2 * (\text{Forward-Time} - 1) \geq \text{Max-Age}$$

$$\text{Max-Age} \geq 2 * (\text{Hello-Time} + 1)$$

### Example

The following example configures the spanning-tree bridge maximum age to 10 seconds.

---

```
Console(config)# spanning-tree max-age 10
```

---

## 29.6 spanning-tree priority

Use the **spanning-tree priority** Global Configuration mode command to configure the device spanning-tree priority, which is used to determine which bridge is selected as the root bridge. Use the **no** form of this command to restore the default device spanning-tree priority.

### Syntax

**spanning-tree priority** *priority*

**no spanning-tree priority**

### Parameters

**priority**—Specifies the bridge priority. (Range: 0–61440)

### Default Configuration

The default bridge priority for IEEE Spanning Tree Protocol (STP) is 32768.

### Command Mode

Global Configuration mode

### User Guidelines

The priority value must be a multiple of 4096.

The switch with the lowest priority is the root of the spanning tree.

### Example

The following example configures the spanning-tree priority to 12288.

---

```
Console(config)# spanning-tree priority 12288
```

---

## 29.7 spanning-tree disable

Use the **spanning-tree disable** Interface Configuration (Ethernet, port-channel) mode command to disable the spanning tree on a specific port. Use the **no** form of this command to enable the spanning tree on a port.

### Syntax

**spanning-tree disable**

**no spanning-tree disable**

### Default Configuration

Spanning tree is enabled on all ports.

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### Example

The following example disables the spanning tree on `gi5`

---

```
Console(config)# interface gi5
Console(config-if)# spanning-tree disable
```

---

## 29.8 spanning-tree cost

Use the **spanning-tree cost** Interface Configuration (Ethernet, port-channel) mode command to configure the spanning-tree path cost for a port. Use the **no** form of this command to restore the default configuration.

### Syntax

**spanning-tree cost** *cost*

**no spanning-tree cost**

### Parameters

**cost**—Specifies the port path cost. (Range: 1–200000000)



### Default Configuration

Default path cost is determined by port speed and path cost method (long or short) as shown below:

Interface	Long	Short
Port-channel	20,000	4
Gigabit Ethernet (1000 Mbps)	20,000	4
Fast Ethernet (100 Mbps)	200,000	19
Ethernet (10 Mbps)	2,000,000	100

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### Example

The following example configures the spanning-tree cost on `gi 15` to 35000.

```
Console(config)# interface gi15
Console(config-if)# spanning-tree cost 35000
```

## 29.9 spanning-tree port-priority

Use the **spanning-tree port-priority** Interface Configuration (Ethernet, port-channel) mode command to configure the port priority. Use the **no** form of this command to restore the default configuration.

### Syntax

**spanning-tree port-priority** *priority*

**no spanning-tree port-priority**

### Parameters

**priority**—Specifies the port priority. (Range: 0–240)

### Default Configuration

The default port priority for IEEE Spanning Tree Protocol (STP) is 128.

**Command Mode**

Interface Configuration (Ethernet, port-channel) mode

**User Guidelines**

The priority value must be a multiple of 16.

**Example**

The following example configures the spanning priority on `gi 15` to `96`

---

```
Console(config)# interface gi15
Console(config-if)# spanning-tree port-priority 96
```

---

**29.10 spanning-tree portfast**

Use the **spanning-tree portfast** Interface Configuration (Ethernet, port-channel) mode command to enable the PortFast mode. In PortFast mode, the interface is immediately put into the forwarding state upon linkup, without waiting for the standard forward time delay. Use the **no** form of this command to disable the PortFast mode.

**Syntax**

**spanning-tree portfast [auto]**

**no spanning-tree portfast**

**Parameters**

**auto**—Specifies that the software waits for 3 seconds (with no BPDUs received on the interface) before putting the interface into the PortFast mode.

**Default Configuration**

PortFast mode is disabled.

**Command Mode**

Interface Configuration (Ethernet, port-channel) mode

### Example

The following example enables the PortFast mode on `gi 15`.

---

```
Console(config)# interface gi15
Console(config-if)# spanning-tree portfast
```

---

## 29.11 spanning-tree link-type

Use the **spanning-tree link-type** Interface Configuration (Ethernet, port-channel) mode command to override the default link-type setting determined by the port duplex mode, and enable Rapid Spanning Tree Protocol (RSTP) transitions to the forwarding state. Use the **no** form of this command to restore the default configuration.

### Syntax

**spanning-tree link-type** *{point-to-point / shared}*

**no spanning-tree spanning-tree link-type**

### Parameters

- **point-to-point**—Specifies that the port link type is point-to-point.
- **shared**—Specifies that the port link type is shared.

### Default Configuration

The device derives the port link type from the duplex mode. A full-duplex port is considered a point-to-point link and a half-duplex port is considered a shared link.

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### Example

The following example enables shared spanning-tree on `gi15`.

---

```
Console(config)# interface gi15
Console(config-if)# spanning-tree link-type shared
```

---

---

## 29.12 spanning-tree pathcost method

Use the **spanning-tree pathcost method** Global Configuration mode command to set the default path cost method. Use the **no** form of this command to return to the default configuration.

### Syntax

**spanning-tree pathcost method** *{long / short}*

**no spanning-tree pathcost method**

### Parameters

- **long**—Specifies that the default port path costs are within the range: 1–200,000,000.
- **short**—Specifies that the default port path costs are within the range: 1–65,535.

### Default Configuration

Long path cost method.

### Command Mode

Global Configuration mode

### User Guidelines

This command applies to all the spanning tree instances on the switch.

- If the short method is chosen, the switch use for the default cost values in the range 1 through 65,535.
- If the long method is chosen, the switch use for the default cost values in the range 1 through 200,000,000.

### Example

The following example sets the default path cost method to Long.

---

```
Console(config)# spanning-tree pathcost method long
```

### 29.13 spanning-tree bpdu (Global)

Use the **spanning-tree bpdu** Global Configuration mode command to define BPDU handling when the spanning tree is disabled globally or on a single interface. Use the **no** form of this command to restore the default configuration.

#### Syntax

**spanning-tree bpdu** {*filtering* / *flooding* / *bridging*}

**no spanning-tree bpdu**

#### Parameters

- **filtering**—Specifies that BPDU packets are filtered when the spanning tree is disabled on an interface.
- **flooding**—Specifies that untagged BPDU packets are flooded unconditionally (without applying VLAN rules) to all ports with the spanning tree disabled and BPDU handling mode of flooding. Tagged BPDU packets are filtered.
- **bridging**—Specifies that BPDU packets, whether untagged or tagged, are flooded and are subject to ingress and egress VLAN rules when the spanning tree is disabled globally. This mode is not relevant if the spanning tree is disabled only on a group of ports.

#### Default Configuration

The default setting is **flooding**.

#### Command Mode

Global Configuration mode

#### User Guidelines

The **filtering** and **flooding** modes are relevant when the spanning tree is disabled globally or on a single interface.

The **bridging** mode is relevant only when the spanning tree is disabled globally.

The BPDU handling mode cannot be changed to **bridging** if the spanning tree is globally enabled.

Conversely, the spanning tree cannot be globally enabled if the BPDU handling mode is **bridging**.

### Example

The following example defines the BPDU packet handling mode as **flooding** when the spanning tree is disabled on an interface.

```
Console(config)# spanning-tree bpdud flooding
```

## 29.14 spanning-tree bpdud (Interface)

Use the **spanning-tree bpdud** Interface Configuration (Ethernet, Port-channel) mode command to define BPDU handling when the spanning tree is disabled on a single interface. Use the **no** form of this command to restore the default configuration.

### Syntax

**spanning-tree bpdud** *{filtering / flooding}*

**no spanning-tree bpdud**

### Parameters

- **filtering**—Specifies that BPDU packets are filtered when the spanning tree is disabled on an interface.
- **flooding**—Specifies that untagged BPDU packets are flooded unconditionally (without applying VLAN rules) to ports with the spanning tree disabled and BPDU handling mode of flooding. Tagged BPDU packets are filtered.

### Default Configuration

The **spanning-tree bpdud (Global)** command determines the default configuration.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

If the **spanning-tree bpdud (Global)** command is supported and the **bridging** mode is supported:

If the global BPDU handling mode is **bridging**, the operational BPDU handling mode is bridging for all the ports (The per-interface BPDU handling configuration is kept as a shadow configuration).

### Example

The following example defines the BPDU packet as **flooding** when the spanning tree is disabled on `gi3`.

---

```
Console(config)# interface gi3
Console(config-if)# spanning-tree bpd flooding
```

---

## 29.15 spanning-tree bpduguard

Use the **spanning-tree bpduguard** Interface Configuration (Ethernet, port-channel) mode command to shut down an interface when it receives a bridge protocol data unit (BPDU). Use the **no** form of this command to restore the default configuration.

### Syntax

**spanning-tree bpduguard** *{enable / disable}*

**no spanning-tree bpduguard**

### Parameters

**enable**—Enables BPDU Guard.

**disable**—Disables BPDU Guard.

### Default Configuration

BPDU Guard is disabled.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

The command can be enabled when the spanning tree is enabled (useful when the port is in the PortFast mode) or disabled.

### Example

The following example shuts down `gi5` when it receives a BPDU.

---

```
Console(config)# interface gi5
```

---

```
Console(config-if)# spanning-tree bpduguard enable
```

---

## 29.16 clear spanning-tree detected-protocols

Use the **clear spanning-tree detected-protocols** Privileged EXEC command to restart the protocol migration process (force the renegotiation with neighboring switches) on all interfaces or on the specified interface

### Syntax

```
clear spanning-tree detected-protocols [interface interface-id]
```

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

### Command Mode

Privileged EXEC mode

### User Guidelines

This feature should be used only when working in RSTP or MSTP mode.

### Example

---

```
console# clear spanning-tree detected-protocols
```

---

## 29.17 spanning-tree mst priority

Use the **spanning-tree mst priority** Global Configuration mode command to configure the device priority for the specified spanning-tree instance. Use the **no** form of this command to restore the default configuration.

### Syntax

```
spanning-tree mst instance-id priority priority
```

```
no spanning-tree mst instance-id priority
```

### Parameters

- **instance-id**—Specifies the spanning-tree instance ID. (Range: 1–7)



- **priority**—Specifies the device priority for the specified spanning-tree instance. This setting affects the likelihood that the switch is selected as the root switch. A lower value increases the probability that the switch is selected as the root switch. (Range: 0–61440)

### Default Configuration

The default bridge priority for IEEE Spanning Tree Protocol (STP) is 32768.

### Command Mode

Global Configuration mode

### User Guidelines

The priority value must be a multiple of 4096.

The switch with the lowest priority is the root of the spanning tree.

### Example

The following example configures the spanning tree priority of instance 1 to 4096.

---

```
Console(config)# spanning-tree mst 1 priority 4096
```

---

## 29.18 spanning-tree mst max-hops

Use the **spanning-tree mst max-hops** Global Configuration mode command to configure the number of hops in an MST region before the BPDU is discarded and the port information is aged out. Use the **no** form of this command to restore the default configuration.

### Syntax

```
spanning-tree mst max-hops hop-count
```

```
no spanning-tree mst max-hops
```

### Parameters

**hop-count**—Specifies the number of hops in an MST region before the BPDU is discarded. (Range: 1–40)

### Default Configuration

The default number of hops is 20.

### Command Mode

Global Configuration mode

### Example

The following example configures the maximum number of hops that a packet travels in an MST region before it is discarded to 10.

---

```
Console(config)# spanning-tree mst max-hops 10
```

---

## 29.19 spanning-tree mst port-priority

Use the **spanning-tree mst port-priority** Interface Configuration (Ethernet, port-channel) mode command to configure the priority of a port. Use the **no** form of this command to restore the default configuration.

### Syntax

```
spanning-tree mst instance-id port-priority priority
```

```
no spanning-tree mst instance-id port-priority
```

### Parameters

- **instance-id**—Specifies the spanning tree instance ID. (Range: 1–15)
- **priority**—Specifies the port priority. (Range: 0–240 in multiples of 16)

### Default Configuration

The default port priority for IEEE Spanning Tree Protocol (STP) is 128.

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### User Guidelines

The priority value must be a multiple of 16.

## Example

The following example configures the port priority of port gi1 to 144.

---

```
Console(config)# interface gi1
Console(config-if)# spanning-tree mst 1 port-priority 144
```

---

## 29.20 spanning-tree mst cost

Use the **spanning-tree mst cost** Interface Configuration (Ethernet, Port-channel) mode command to configure the path cost for multiple spanning-tree (MST) calculations. If a loop occurs, the spanning tree considers path cost when selecting an interface to put in the forwarding state. Use the **no** form of this command to restore the default configuration.

### Syntax

**spanning-tree mst** *instance-id* **cost** *cost*

**no spanning-tree mst** *instance-id* **cost**

### Parameters

- **instance-id**—Specifies the spanning-tree instance ID. (Range: 1–15)
- **cost**—Specifies the port path cost. (Range: 1–200000000)

### Default Configuration

Default path cost is determined by the port speed and path cost method (long or short) as shown below:

Interface	Long	Short
Port-channel	20,000	4
Gigabit Ethernet (1000 Mbps)	20,000	4
Fast Ethernet (100 Mbps)	200,000	19
Ethernet (10 Mbps)	2,000,000	100

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### Example

The following example configures the MSTP instance 1 path cost for gigabitethernet port 9 to 4.

```
Console(config)# interface gi9
Console(config-if)# spanning-tree mst 1 cost 4
```

---

## 29.21 spanning-tree mst configuration

Use the **spanning-tree mst configuration** Global Configuration mode command to enable configuring an MST region by entering the Multiple Spanning Tree (MST) mode.

### Syntax

**spanning-tree mst configuration**

### Command Mode

Global Configuration mode

### User Guidelines

For two or more switches to be in the same MST region, they need to contain the same VLAN mapping, the same configuration revision number, and the same name.

### Example

The following example configures an MST region.

```
Console(config)# spanning-tree mst configuration
Console(config-mst)# instance 1 vlan 10-20
Console(config-mst)# name region1
Console(config-mst)# revision 1
```

---

## 29.22 instance (MST)

Use **instance** MST Configuration mode command to map VLANs to an MST instance. Use the **no** form of this command to restore default mapping.

### Syntax

**instance** *instance-id* **vlan** *vlan-range*

**no instance** *instance-id* **vlan** *vlan-range*

### Parameters

- **instance-id**—MST instance (Range: 1–16)
- **vlan-range**—The specified range of VLANs is added to the existing ones. To specify a range, use a hyphen. To specify a series, use a comma. (Range: 1–4094)

### Default Configuration

All VLANs are mapped to the common and internal spanning tree (CIST) instance (instance 0).

### Command Mode

MST Configuration mode

### User Guidelines

All VLANs that are not explicitly mapped to an MST instance are mapped to the common and internal spanning tree (CIST) instance (instance 0) and cannot be unmapped from the CIST.

For two or more devices to be in the same MST region, they must have the same VLAN mapping, the same configuration revision number, and the same name.

### Example

The following example maps VLANs 10-20 to MST instance 1.

---

```
Console(config)# spanning-tree mst configuration
Console(config-mst)# instance 1 vlan 10-20
```

---

### 29.23 name (MST)

Use the **name** MST Configuration mode command to define the MST configuration name. Use the **no** form of this command to restore the default setting.

#### Syntax

**name** *string*

**no name**

#### Parameters

**string**—Specifies the MST configuration name. (Length: 1–32 characters)

#### Default Configuration

The default name is the bridge MAC address.

#### Command Mode

MST Configuration mode

#### Example

The following example defines the configuration name as Region1.

---

```
Console(config)# spanning-tree mst configuration
```

```
Console(config-mst)# name region1
```

---

### 29.24 revision (MST)

Use the **revision** MST Configuration mode command to define the MST configuration revision number. Use the **no** form of this command to restore the default configuration.

#### Syntax

**revision** *value*

**no revision**

**Parameters**

**value**—Specifies the MST configuration revision number. (Range: 0–65535)

**Default Configuration**

The default configuration revision number is 0.

**Command Mode**

MST Configuration mode

**Example**

The following example sets the configuration revision to 1.

---

```
Console(config) # spanning-tree mst configuration
Console(config-mst) # revision 1
```

---

**29.25 show (MST)**

Use the **show MST Configuration mode** command to displays the current or pending MST region configuration.

**Syntax**

**show** {*current* / *pending*}

**Parameters**

- **current**—Displays the current MST region configuration.
- **pending**—Displays the pending MST region configuration.

**Command Mode**

MST Configuration mode

### Example

The following example displays a pending MST region configuration.

---

```
Console(config-mst)# show pending
Pending MST configuration
Name: Region1
Revision: 1
Instance      VLANs Mapped      State
-----      -
0             1-9,21-4094      Enabled
1             10-20             Enabled
```

---

### 29.26 exit (MST)

Use the **exit** MST Configuration mode command to exit the MST region Configuration mode and apply all configuration changes.

#### Syntax

**exit**

#### Command Mode

MST Configuration mode

#### Example

The following example exits the MST Configuration mode and saves changes.

---

```
Console(config)# spanning-tree mst configuration
Console(config-mst)# exit
```

---

### 29.27 abort (MST)

Use the **abort** MST Configuration mode command to exit the MST Configuration mode without applying the configuration changes.



**Syntax****abort****Command Mode**

MST Configuration mode

**Example**

The following example exits the MST Configuration mode without saving changes.

---

```
Console(config)# spanning-tree mst configuration
```

```
Console(config-mst)# abort
```

---

**29.28 show spanning-tree**

Use the **show spanning-tree** Privileged EXEC mode command to display the spanning-tree configuration.

**Syntax****show spanning-tree** [*interface-id*] [*instance instance-id*]**show spanning-tree** [*detail*] [*active | blockedports*] [*instance instance-id*]**show spanning-tree** *mst-configuration***Parameters**

- **instance** *instance-id*—Specifies the spanning tree instance ID. (Range: 1–16)
- **detail**—Displays detailed information.
- **active**—Displays active ports only.
- **blockedports**—Displays blocked ports only.
- **mst-configuration**—Displays the MST configuration identifier.
- **interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Command Mode**

Privileged EXEC mode

### Example

The following examples display spanning-tree information.`show spanning-tree`

---

```
Console# show spanning-tree

Spanning tree enabled mode RSTP

Default port cost method: long

Loopback guard: Disabled

Root ID   Priority           32768
         Address          00:01:42:97:e0:00
         Cost            20000
         Port            gil
         Hello Time 2 sec      Max Age 20      Forward Delay 15
                               sec                  sec

Bridge ID Priority           36864
         Address          00:02:4b:29:7a:00
         Hello Time 2 sec      Max Age 20      Forward Delay 15
                               sec                  sec
```

```

Interfaces
Name      State  Prio. No  Cost  Sts  Role  PortFas Type
-----  -----  -----  -----  ---  ----  -
gi1      Enabled 128.1    20000 FWD  Root  ----- P2p (RSTP)
gi2      Enabled 128.2    20000 FWD  Desg  No     Shared
gi3      Disable 128.3    20000 -     -     No     (STP)
gi4      d       128.4    20000 BLK  Altn  -     -
gi5      Enabled 128.5    20000 DIS  -     No     Shared
          Enabled                    20000 DIS  -     (STP)
          Enabled                    20000 DIS  -     -

```

---

```

Console# show spanning-tree

```

```

Spanning tree enabled mode RSTP

```

```

Default port cost method: long

```

```

Root ID  Priority          36864
        Address          00:02:4b:29:7a:00
        This switch is the Root.
        Hello Time 2 sec          Max Age 20          Forward Delay 15 sec
                                   sec

```

```

Interfaces

```

Name	State	Prio.Nbr	Cost	Sts	Role	PortFas	Type
-----	-----	-----	-----	---	----	t	-----
gi1	-	128.1	20000	FWD	Desg	-----	P2p (RSTP)
gi2	Enabled	128.2	20000	FWD	Desg	--	Shared
gi3	Enabled	128.3	20000	-	-	No	(STP)
gi4	Disable	128.4	20000	FWD	Desg	No	-
gi5	d	128.5	20000	DIS	-	-	Shared
	Enabled					No	(STP)
	Enabled					-	-

Console# **show spanning-tree**

Spanning tree disabled (BPDU filtering) mode RSTP

Default port cost method: long

Root ID	Priority	N/A			
	Address	N/A			
	Path Cost	N/A			
	Root Port	N/A			
	Hello Time	N/A	Max Age	N/A	Forward Delay N/A

Bridge ID	Priority	36864			
	Address	00:02:4b:29:7a:00			
	Hello Time	2 sec	Max Age	20	Forward Delay 15 sec
			sec		

Interfaces

Name	State	Prio.Nb	Cost	Sts	Role	PortFas	Type
-----	-----	-----	-----	---	----	t	-----
gi1	-	128.1	20000	-	-	-----	-
gi2	Enabled	128.2	20000	-	-	--	-
gi3	Enabled	128.3	20000	-	-	-	-
gi4	Disable	128.4	20000	-	-	-	-
gi5	d	128.5	20000	-	-	-	-
	Enabled					-	
	Enabled					-	

Console# **show spanning-tree active**

Spanning tree enabled mode RSTP

Default port cost method: long

```

Root ID   Priority           32768
          Address         00:01:42:97:e0:00
          Path Cost      20000
          Root Port     gi1
          Hello Time 2 sec           Max Age 20 sec Forward Delay 15
                                   sec

```

```

Bridge ID Priority           36864
          Address         00:02:4b:29:7a:00
          Hello Time 2 sec           Max Age 20 sec Forward Delay 15
                                   sec

```

Interfaces

Name	State	Prio.Nbr	Cost	Sts	Role	PortFas	Type
-----	-----	-----	-----	---	----	t	-----
gi1	-	128.1	20000	FWD	Root	-----	P2p (RSTP)
gi2	Enabled	128.2	20000	FWD	Desg	--	Shared
gi4	Enabled	128.4	20000	BLK	Altn	No	(STP)
	Enabled					No	Shared
						No	(STP)

Console# **show spanning-tree blockedports**

Spanning tree enabled mode RSTP

Default port cost method: long

```

Root ID   Priority           32768
          Address           00:01:42:97:e0:00
          Path Cost        20000
          Root Port       gi1
          Hello Time 2 sec           Max Age 20 sec           Forward Delay 15 sec

```

```

Bridge ID Priority           36864
          Address           00:02:4b:29:7a:00
          Hello Time 2 sec           Max Age 20 sec           Forward Delay 15 sec

```

Interfaces

Name	State	Prio.Nbr	Cost	Sts	Role	PortFas	Type
-----	-----	-----	-----	---	----	t	-----
gi4	Enabled	128.4	19	BLK	Altn	-----	Shared
						-	(STP)
						No	

---

```
Console# show spanning-tree detail
```

```
Spanning tree enabled mode RSTP
```

```
Default port cost method: long
```

```

Root ID   Priority           32768
        Address           00:01:42:97:e0:00
        Path Cost        20000
        Root Port        gi1
        Hello Time 2 sec           Max Age 20           Forward Delay 15
                                   sec                       sec

```

```

Bridge ID Priority           36864
        Address           00:02:4b:29:7a:00
        Hello Time 2 sec           Max Age 20           Forward Delay 15
                                   sec                       sec

```

```
Number of topology changes 2 last change occurred 2d18h ago
```

```
Times: hold 1, topology change 35, notification 2
```

```
hello 2, max age 20, forward delay 15
```

```
Port 1 (gi1) enabled
```

```
State: Forwarding           Role: Root
```

```
Port id: 128.1           Port cost: 20000
```

```
Type: P2p (configured: auto) RSTP Port Fast: No (configured:no)
```

```
Designated bridge Priority: 32768 Address: 00:01:42:97:e0:00
```

```
Designated port id: 128.25           Designated path cost: 0
```

```
Guard root: Disabled           BPDU guard: Disabled
```

```
Number of transitions to forwarding state: 1
BPDU: sent 2, received 120638

Port 2 (gi2) enabled
State: Forwarding                               Role: Designated
Port id: 128.2                                   Port cost: 20000
Type: Shared (configured: auto)                 Port Fast: No (configured:no)
STP                                              Address: 00:02:4b:29:7a:00
Designated bridge Priority: 32768               Designated path cost: 20000
Designated port id: 128.2                       BPDU guard: Disabled
Guard root: Disabled
Number of transitions to forwarding state: 1
BPDU: sent 2, received 170638

Port 3 (gi3) disabled
State: N/A                                       Role: N/A
Port id: 128.3                                   Port cost: 20000
Type: N/A (configured: auto)                   Port Fast: N/A (configured:no)
Designated bridge Priority: N/A                 Address: N/A
Designated port id: N/A                         Designated path cost: N/A
Guard root: Disabled                           BPDU guard: Disabled
Number of transitions to forwarding state: N/A
BPDU: sent N/A, received N/A

Port 4 (gi4) enabled
State: Blocking                                 Role: Alternate
Port id: 128.4                                   Port cost: 20000
Type: Shared (configured:auto) STP Port Fast: No (configured:no)
Designated bridge Priority: 28672 Address: 00:30:94:41:62:c8
Designated port id: 128.25                       Designated path cost: 20000
Guard root: Disabled                           BPDU guard: Disabled
```



---

```
Number of transitions to forwarding state: 1
BPDU: sent 2, received 120638

Port 5 (gi5) enabled
State: Disabled                               Role: N/A
Port id: 128.5                                Port cost: 20000
Type: N/A (configured: auto)                 Port Fast: N/A (configured:no)
Designated bridge Priority: N/A              Address: N/A
Designated port id: N/A                      Designated path cost: N/A
Guard root: Disabled                         BPDU guard: Disabled
Number of transitions to forwarding state: N/A
BPDU: sent N/A, received N/A
```

---

---

```

Console# show spanning-tree ethernet gil
Port 1 (gil) enabled

State: Forwarding                Role: Root
Port id: 128.1                    Port cost: 20000
Type: P2p (configured: auto) RSTP Port Fast: No (configured:no)
Designated bridge Priority: 32768 Address: 00:01:42:97:e0:00
Designated port id: 128.25        Designated path cost: 0
Guard root: Disabled              BPDU guard: Disabled
Number of transitions to forwarding state: 1
BPDU: sent 2, received 120638

```

---

```

Console# show spanning-tree mst-configuration

```

```

Name: Region1
Revision: 1

```

Instance	Vlans mapped	State
0	1-9, 21-4094	Enabled
1	10-20	Enabled

---

```

Console# show spanning-tree

```

```

Spanning tree enabled mode MSTP

```

```

Default port cost method: long

```

```

##### MST 0 Vlans Mapped: 1-9

```

```

CST Root ID      Priority  32768
                  Address   00:01:42:97:e0:00
                  Path Cost 20000
                  Root Port  gil
                  Hello Time 2 sec Max Age 20 Forward Delay 15
                               sec          sec

```

```

IST Master ID    Priority  32768
                 Address   00:02:4b:29:7a:00
                 This switch is the IST master.
                 Hello Time 2 sec Max Age 20 Forward Delay 15
                               sec          sec
                 Max hops 20

```

## Interfaces

Name	State	Prio.Nbr	Cost	Sts	Role	PortFas	Type
----	-----	-----	-----	---	----	t	-----
gi1	Enabled	128.1	20000	FWD	Root	-----	-
gi2	Enabled	128.2	20000	FWD	Desg	-	P2p Bound (RSTP)
gi3	Enabled	128.3	20000	FWD	Desg	No	Shared
gi4	Enabled	128.4	20000	FWD	Desg	No	Bound (STP)
						No	P2p
							P2p

```
##### MST 1 Vlans Mapped: 10-20
```

```

Root ID          Priority  24576
                 Address   00:02:4b:29:89:76
                 Path Cost 20000
                 Root Port  gi4
                 Rem hops  19

```

```

Bridge ID        Priority  32768
                 Address   00:02:4b:29:7a:00

```

```

Interfaces
Name      State  Prio.Nbr  Cost  Sts  Role  PortFas Type
-----  -
gi1      Enabled 128.1   20000  FWD  Boun  -
gi2      Enabled 128.2   20000  FWD  Boun  -      P2p Bound
gi3      Enabled 128.3   20000  BLK  Altn  No     (RSTP)
gi4      Enabled 128.4   20000  FWD  Root  No     Shared
                                                No     Bound (STP)
                                                No     P2p
                                                P2p

```

```

Console# show spanning-tree detail

```

```

Spanning tree enabled mode MSTP

```

```

Default port cost method: long

```

```

##### MST 0 Vlans Mapped: 1-9

```

```

CST Root ID      Priority  32768
                  Address   00:01:42:97:e0:00
                  Path Cost 20000
                  Root Port  gi1
                  Hello Time 2 sec Max Age 20 Forward Delay 15
                               sec          sec

```

```

IST Master ID    Priority  32768
                  Address   00:02:4b:29:7a:00
                  This switch is the IST master.
                  Hello Time 2 sec Max Age 20 Forward Delay 15
                               sec          sec

```

```
Max hops 20
Number of topology changes 2 last change occurred
2d18h ago
Times: hold 1, topology change 35, notification 2
hello 2, max age 20, forward delay 15
```

```
Port 1 (gi1) enabled
```

```
State: Forwarding                Role: Root
Port id: 128.1                    Port cost: 20000
Type: P2p (configured: auto) Boundary Port Fast: No
RSTP                               (configured:no)
Designated bridge Priority: 32768   Address: 00:01:42:97:e0:00
Designated port id: 128.25         Designated path cost: 0
Number of transitions to forwarding
state: 1
BPDU: sent 2, received 120638
```

```
Port 2 (gi2) enabled
```

```
State: Forwarding                Role: Designated
Port id: 128.2                    Port cost: 20000
Type: Shared (configured: auto) Boundary Port Fast: No
STP                               (configured:no)
Designated bridge Priority: 32768   Address: 00:02:4b:29:7a:00
Designated port id: 128.2         Designated path cost:
Number of transitions to forwarding 20000
state: 1
BPDU: sent 2, received 170638
```

```
Port 3 (gi3) enabled
State: Forwarding                               Role: Designated
Port id: 128.3                                  Port cost: 20000
Type: Shared (configured: auto) Internal Port Fast: No
Designated bridge Priority: 32768              (configured:no)
Designated port id: 128.3                     Address: 00:02:4b:29:7a:00
Number of transitions to forwarding           Designated path cost:
state: 1                                       20000
BPDU: sent 2, received 170638

Port 4 (gi4) enabled
State: Forwarding                               Role: Designated
Port id: 128.4                                  Port cost: 20000
Type: Shared (configured: auto) Internal Port Fast: No
Designated bridge Priority: 32768              (configured:no)
Designated port id: 128.2                     Address: 00:02:4b:29:7a:00
Number of transitions to forwarding           Designated path cost:
state: 1                                       20000
BPDU: sent 2, received 170638

##### MST 1 Vlans Mapped: 10-20
Root ID      Priority  24576
              Address   00:02:4b:29:89:76
              Path Cost 20000
              Root Port  gi4
              Rem hops 19

Bridge ID    Priority  32768
              Address   00:02:4b:29:7a:00
              Number of topology changes 2 last change occurred
              1d9h ago
```

```
Times: hold 1, topology change 2, notification 2
hello 2, max age 20, forward delay 15
```

```
Port 1 (gi1) enabled
```

```
State: Forwarding                      Role: Boundary
Port id: 128.1                          Port cost: 20000
Type: P2p (configured: auto) Boundary   Port Fast: No
RSTP                                     (configured:no)
Designated bridge Priority: 32768       Address: 00:02:4b:29:7a:00
Designated port id: 128.1              Designated path cost:
Number of transitions to forwarding     20000
state: 1
BPDU: sent 2, received 120638
```

```
Port 2 (gi2) enabled
```

```
State: Forwarding                      Role: Designated
Port id: 128.2                          Port cost: 20000
Type: Shared (configured: auto) Boundary Port Fast: No
STP                                     (configured:no)
Designated bridge Priority: 32768       Address: 00:02:4b:29:7a:00
Designated port id: 128.2              Designated path cost:
Number of transitions to forwarding     20000
state: 1
BPDU: sent 2, received 170638
```

```

Port 3 (gi3) disabled
State: Blocking                               Role: Alternate
Port id: 128.3                                Port cost: 20000
Type: Shared (configured: auto) Internal Port Fast: No
Designated bridge Priority: 32768             (configured:no)
Designated port id: 128.78                   Address: 00:02:4b:29:1a:19
Number of transitions to forwarding          Designated path cost:
state: 1                                     20000
BPDU: sent 2, received 170638

```

```

Port 4 (gi4) enabled
State: Forwarding                             Role: Designated
Port id: 128.4                                Port cost: 20000
Type: Shared (configured: auto) Internal Port Fast: No
Designated bridge Priority: 32768             (configured:no)
Designated port id: 128.2                   Address: 00:02:4b:29:7a:00
Number of transitions to forwarding          Designated path cost:
state: 1                                     20000
BPDU: sent 2, received 170638

```

---

```

Console# show spanning-tree

```

```

Spanning tree enabled mode MSTP

```

```

Default port cost method: long

```

```

##### MST 0 Vlans Mapped: 1-9

```

```

CST Root ID      Priority  32768
                  Address   00:01:42:97:e0:00
                  Path Cost 20000
                  Root Port  gi1
                  Hello Time 2 sec Max Age 20   Forward Delay 15
                                 sec           sec

```



```

IST Master ID    Priority  32768
                 Address   00:02:4b:19:7a:00
                 Path Cost 10000
                 Rem hops  19

Bridge ID       Priority  32768
                 Address   00:02:4b:29:7a:00
                 Hello Time 2 sec Max Age 20   Forward Delay 15
                                   sec           sec
                 Max hops  20

```

---

```

Console# show spanning-tree

```

```

Spanning tree enabled mode MSTP

```

```

Default port cost method: long

```

```

##### MST 0 Vlans Mapped: 1-9

```

```

CST Root ID     Priority  32768
                 Address   00:01:42:97:e0:00
                 This switch is root for CST and IST master.
                 Root Port  gil
                 Hello Time 2 sec Max Age 20   Forward Delay 15
                                   sec           sec
                 Max hops  20

```

```

bpdud

```

---

## 29.29 show spanning-tree bpdud

Use the **show spanning-tree bpdud** EXEC mode command to display the BPDU handling when spanning-tree is disabled.

### Syntax

```
show spanning-tree bpdud [interface-id]
```

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

## Command Mode

EXEC mode

## Example

The following examples display spanning-tree information:

---

```
Console# show spanning-tree bpdu
```

```
The following is the output if the global BPDU handling
command is not supported.
```

```
The following is the output if both the global BPDU
handling command and the per-interface BPDU handling
command are supported.
```

```
Global: Flooding
```

Interface	Admin Mode	Oper Mode
-----	-----	-----
gi1	Global	Flooding
gi2	Global	STP
gi3	Flooding	STP

```
The following is the output if bridging mode is supported.
```

---

## 30 Virtual Local Area Network (VLAN) Commands

---

### 30.1 `vlan database`

Use the **vlan database** Global Configuration mode command to enter the VLAN Configuration mode. Commands in this mode are at the VLAN level and perform actions, such as creating and naming VLANs and defining the default VLAN.

Use the **exit** command to return to Global Configuration mode.

#### Syntax

**vlan database**

#### Parameters

N/A

#### Default Configuration

VLAN 1 exists by default.

#### Command Mode

Global Configuration mode

#### Example

The following example enters the VLAN Configuration mode, creates VLAN 1972 and exits VLAN Configuration mode.

---

```
Console(config)# vlan database
Console(config-vlan)# vlan 1972
Console(config-vlan)# exit
Console(config)#
```

---

### 30.2 `vlan`

Use the **vlan** VLAN Configuration mode command to create a VLAN. Use the **no** form of this command to delete the VLAN(s). To assign the VLAN a name, use **name**.

### Syntax

**vlan** *vlan-range*

**no vlan** *vlan-range*

### Parameters

- **vlan-range**—Specifies a list of VLAN IDs to add. Separate nonconsecutive VLAN IDs with a comma and no spaces. Use a hyphen to designate a range of IDs (range: 2-4094).

### Default Configuration

VLAN 1 exists by default.

### Command Mode

VLAN Configuration mode

### Example

The following example creates VLAN number 1972.

---

```
Console(config)# vlan database
```

```
Console(config-vlan)# vlan 1972
```

```
Console(config-vlan)#
```

---

## 30.3 show vlan

Use the **show vlan** Privileged EXEC mode command to display the following VLAN information for all VLANs or for a specific VLAN:

- VLAN ID
- VLAN name
- Ports on the VLAN
- Whether the VLAN was created dynamically or statically
- Whether authorization is required on the VLAN

**Syntax**

**show vlan** [*tag vlan-id / name vlan-name*]

**Parameters**

- **tag** *vlan-id*—Specifies a VLAN ID.
- **name** *vlan-name*—Specifies a VLAN name string (length: 1–32 characters)

**Default Configuration**

All VLANs are displayed.

**Command Mode**

Privileged EXEC mode

**Examples:**

**Example 1** - The following example displays information for all VLANs:

---

```

Console# show vlan

VLAN  Name          Ports          Type          Authorization
----  -
1      default         gi1-2          Default       Required
10     Marketing       gi3-14         static        Required
11     VLAN0011        gi5-16         static        Required
20     VLAN0020        gi7-18         static        Required
21     VLAN0021                            static        Required
30     VLAN0030                            static        Required
31     VLAN0031                            static        Required
91     VLAN0091        gi2            dynamic       Not Required
3978   Guest          gi7            static        Guest
VLAN

```

---

**Example 2** - The following example displays information for the default VLAN (VLAN 1):

---

```
Console# show vlan tag default
```

VLAN	Name	Ports	Type	Authorization
1	default	gi1-2	Default	Required

---

**Example 3** - The following example displays information for the VLAN named Marketing:

---

```
Console# show vlan name Marketing
```

VLAN	Name	Ports	Type	Authorization
1	Marketing	gi3-14	static	Required

---

## 30.4 default-vlan vlan

Use the **default-vlan vlan** VLAN Configuration mode command to define the default VLAN. Use the **no** form of this command to set VLAN 1 as the default VLAN.

### Syntax

**default-vlan vlan** *vlan-id*

**no default-vlan vlan**

### Parameters

**vlan** *vlan-id*—Specifies the default VLAN ID.

### Default Configuration

The default VLAN is 1 by default.

### Command Mode

VLAN Configuration mode

### User Guidelines

This command becomes effective after reboot of the device.

### Example

The following example defines the default VLAN as 2.

---

```
Console(config)# vlan database
```

```
Console(config-vlan)# default-vlan vlan 2
```

New Default VLAN ID will be active after save configuration and reboot device.

---

## 30.5 show default-vlan-membership

Use the **show default-vlan-membership** privileged EXEC command to view the default VLAN membership.

### Syntax

```
show default-vlan-membership [interface-id]
```

### Parameters

**interface-id**—Specify an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel

### Default Configuration

Membership in the default VLAN is displayed for all interfaces.

### Command Mode

Privileged EXEC

### Example

```
Console # show default-vlan-membership
```

```
Port      Forbidden      Membership
-----
gi1       TRUE           FALSE
gi2       FALSE          TRUE
gi3       FALSE          FALSE
```

---

## 30.6 interface vlan

Use the **interface vlan** Global Configuration mode command to enter the Interface Configuration (VLAN) mode for a specific VLAN. After this command is entered, all commands configure this VLAN. To configure a range of VLANs, use **interface range vlan**.

### Syntax

```
interface vlan vlan-id
```

### Parameters

**vlan** *vlan-id*—Specifies the VLAN to be configured.

### Default Configuration

N/A

### Command Mode

Global Configuration mode

### User Guidelines

If the VLAN does not exist (ghost VLAN), some commands are not available under the interface VLAN context.

The commands that are supported for VLANs but do not exist for ghost VLANs are:

- IGMP snooping control commands
- Bridge Multicast configuration commands

### Example

The following example configures VLAN 1 with IP address 131.108.1.27 and subnet mask 255.255.255.0.

---

```
Console (config)# interface vlan 1
Console (config-if)# ip address 131.108.1.27 255.255.255.0
```



---

## 30.7 interface range vlan

Use the **interface range vlan** Global Configuration mode command to configure multiple VLANs simultaneously.

### Syntax

```
interface range vlan vlan-range
```

### Parameters

**vlan** *vlan-range*—Specifies a list of VLANs. Separate nonconsecutive VLANs with a comma and no spaces. Use a hyphen to designate a range of VLANs.

### Default Configuration

N/A

### Command Mode

Global Configuration mode

### User Guidelines

Commands under the interface VLAN range context are executed independently on each VLAN in the range. If the command returns an error on one of the VLANs, an error message is displayed, and the system attempts to configure the remaining VLANs.

If a VLAN does not exist (ghost VLAN), some commands are not available under the interface VLAN context. These are:

- IGMP snooping control commands
- Bridge Multicast configuration commands

### Example

The following example groups VLANs 221 through 228 and 889 to receive the same command(s).

---

```
Console(config)# interface range vlan 221-228, vlan 889  
Console(config-if)#
```

---

## 30.8 name

Use the **name** Interface Configuration (VLAN) mode command to name a VLAN. Use the **no** form of this command to remove the VLAN name. This is the same as using the **vlan** command with the **name** parameter.

### Syntax

**name** *string*

**no name**

### Parameters

**string**—Specifies a unique name associated with this VLAN. (Length: 1–32 characters)

### Default Configuration

No name is defined.

### Command Mode

Interface Configuration (VLAN) mode. It cannot be configured for a range of interfaces (range context).

### User Guidelines

The VLAN name must be unique.

### Example

The following example assigns VLAN 19 the name Marketing.

---

```
Console(config)# interface vlan 19
Console(config-if)# name Marketing
```

---

## 30.9 switchport protected-port

Use the **switchport protected-port** Interface Configuration mode command to isolate Unicast, Multicast, and Broadcast traffic at Layer 2 from other protected ports on the same switch. Use the **no** form of this command to disable protection on the port.

**Syntax****switchport protected-port****no switchport protected-port****Parameters**

N/A

**Default Configuration**

Unprotected

**Command Mode**

Interface configuration (Ethernet, port-channel)

**User Guidelines**

Note that packets are subject to all filtering rules and Filtering Database (FDB) decisions.

**Example**

```
console(config)# interface gil
console(config-if)# switchport protected-port
```

---

**30.10 show interfaces protected-ports**

Use the **show interfaces protected-ports** EXEC mode command to display protected ports configuration.

**Syntax****show interfaces protected-ports** [*interface-id*]**Parameters**

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Default Configuration**

Show all protected ports.

## Command Mode

EXEC mode

## Example

```
console# show interfaces protected-ports
```

Interface	State
-----	-----
gi1	Protected
gi2	Protected
gi3	Unprotected
gi4	Unprotected

## 30.11 switchport mode

Use the **switchport mode** Interface Configuration (Ethernet, port-channel) mode command to configure the VLAN membership mode (access, trunk, general or customer) of a port. Use the **no** form of this command to restore the default configuration.

### Syntax

```
switchport mode {access / trunk / general / customer}
```

```
no switchport mode
```

### Parameters

- **access**—Specifies an untagged layer 2 VLAN port.
- **trunk**—Specifies a trunking layer 2 VLAN port.
- **general**—Specifies a full 802-1q-supported VLAN port.
- **customer**—Specifies that the port is connected to customer equipment. Used when the switch is in a provider network.

### Default Configuration

Trunk mode.

## Command Mode

Interface Configuration (Ethernet, port-channel) mode

## User Guidelines

- When the port's mode is changed, it receives the configuration corresponding to the mode.
- If the port mode is changed to access and the access VLAN does not exist, then the port does not belong to any VLAN.
- Trunk and general mode ports can be changed to access mode only if all VLANs (except for an untagged PVID are first removed.

## Example

The following example configures `gi1` as an access port (untagged layer 2) VLAN port.

```
Console(config)# interface gi1
Console(config-if)# switchport mode access
Console(config-if)# switchport access vlan 2
```

## 30.12 switchport access vlan

An interface in access mode can belong to only one VLAN. The **switchport access vlan** Interface Configuration command reassigns an interface to a different VLAN than it currently belongs to.

Use the **no** form of this command to restore the default configuration.

### Syntax

```
switchport access vlan vlan-id
```

```
no switchport access vlan
```

### Parameters

**vlan** *vlan-id*—Specifies the VLAN ID to which the port is configured.

### Default Configuration

The interface belongs to the default VLAN.

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### User Guidelines

The command automatically removes the port from its previous VLAN and adds it to the new VLAN.

### Example

The following example sets gi1 as an access port and assigns it to VLAN 2 (and removes it from its previous VLAN).

```
Console(config)# interface gi2
Console(config-if)# switchport mode access
Console(config-if)# switchport access vlan 2
```

---

## 30.13 switchport trunk allowed vlan

A trunk interface is an untagged member of a single VLAN, and, in addition, it may be an tagged member of one or more VLANs. The **switchport trunk allowed vlan** Interface Configuration mode command adds/removes VLAN(s) to/from a trunk port.

### Syntax

```
switchport trunk allowed vlan {add vlan-list | remove vlan-list}
```

### Parameters

- **add** *vlan-list* — Specifies a list of VLAN IDs to add to a port. Separate nonconsecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs.
- **remove** *vlan-list* — Specifies a list of VLAN IDs to remove from a port. Separate nonconsecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs.

### Default Configuration

By default, trunk ports belongs to all created VLANs.

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### Example

To add VLANs 2,3 and 100 to trunk ports 1 to 13:

```
console(config)# interface range gi1-13
console(config-if)# switchport mode trunk
console(config-if)# switchport trunk allowed vlan add 2-3,100
console(config-if)#
```

---

## 30.14 switchport trunk native vlan

If an untagged packet arrives on a trunk port, it is directed to the port's native VLAN. Use the **switchport trunk native vlan** Interface Configuration (Ethernet, port-channel) mode command to define the native VLAN for a trunk interface. Use the **no** form of this command to restore the default native VLAN.

### Syntax

**switchport trunk native vlan** *vlan-id*

**no switchport trunk native vlan**

### Parameters

- **vlan-id**—Specifies the native VLAN ID.

### Default Configuration

The default VLAN is the native VLAN.

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### User Guidelines

The command adds the port as a member of the VLAN. If the port is already a member of the VLAN (not a native), it must first be removed from the VLAN.

**Examples:****Example 1** - The following example:

- Defines VLAN 2 as native VLAN for port 1
- Removes VLAN 2 from port 1 and then sets it as the native VLAN

---

```
console(config)# interface gil
console(config-if)# switchport trunk native vlan 2
Port 1: Port is Trunk in VLAN 2.
console(config-if)# switchport trunk allowed vlan remove 2
console(config-if)# switchport trunk native vlan 2
console(config-if)#
```

---

**Example 2** - The following example sets packets on port as untagged on ingress and untagged on egress:

---

```
console(config)# interface gil
console(config-if)# switchport mode trunk
console(config-if)# switchport trunk native vlan 2
console(config-if)#
```

---

**Example 3** - The following example sets packets on port as tagged on ingress and tagged on egress:

---

```
console(config)# interface gil
console(config-if)# switchport mode trunk
console(config-if)# switchport trunk allowed vlan add 2
console(config-if)#
```

---

## 30.15 switchport general allowed vlan

General ports can receive tagged or untagged packets. Use the **switchport general allowed vlan** Interface Configuration mode command to add/remove



VLANs to/from a general port and configure whether packets on the egress are tagged or untagged. Use the **no** form of this command to reset to the default.

### Syntax

**switchport general allowed vlan** {[**add** *vlan-list* [**tagged**| **untagged**]] | [**remove** *vlan-list*]}

### Parameters

- **add** *vlan-list* — Specifies the list of VLAN IDs to add. Separate nonconsecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs.
- **tagged** — Specifies that the port transmits tagged packets for the VLANs. This is the default value
- **untagged** — Specifies that the port transmits untagged packets for the VLANs.
- **remove** *vlan-list* — Specifies the list of VLAN IDs to remove. Separate nonconsecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs.

### Default Configuration

The port is not member in any VLAN.

Packets are transmitted untagged.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

You can change the egress rule (for example, from tagged to untagged) without first removing the VLAN from the list.

### Example

Sets port 1 to general mode and adds VLAN 2 and 3 to it. Packets are tagged on the egress.

```
console(config)# interface gil  
console(config-if)# switchport mode general
```

```
console(config-if)# switchport general allowed vlan add 2-3 tagged
```

---

## 30.16 switchport general pvid

The port VLAN ID (PVID) is the VLAN to which incoming untagged and priority-tagged frames are classified on a general port. Use the **switchport general pvid** Interface Configuration (Ethernet, Port-channel) mode command to configure the Port VLAN ID (PVID) of an interface when it is in general mode. Use the **no** form of this command to restore the default configuration.

### Syntax

```
switchport general pvid vlan-id
```

```
no switchport general pvid
```

### Parameters

**pvid** *vlan-id*—Specifies the Port VLAN ID (PVID).

### Default Configuration

The default VLAN is the PVID.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### Examples

**Example 1** - The following example configures port 2 as a general port and sets its PVID to 234.

---

```
Console(config)# interface gi2  
Console(config-if)# switchport mode general  
Console(config-if)# switchport general pvid 234
```

---

**Example 2** - Performs the following:

- Adds VLANs 2&3 as tagged, and VLAN 100 as untagged to general mode port 14
- Defines VID 100 as the PVID

- Reverts to the default PVID (VID=1)

---

```
console(config)# interface gi14
console(config-if)# switchport mode general
console(config-if)# switchport general allowed vlan add 2-3 tagged
console(config-if)# switchport general allowed vlan add 100 untagged
console(config-if)# switchport general pvid 100
console(config-if)# no switchport general pvid
console(config-if)#
```

---

**Example 3** - Configures VLAN on port 14 as untagged on input and untagged on output:

---

```
console(config)# interface gi14
console(config-if)# switchport mode general
console(config-if)# switchport general pvid 2
console(config-if)# switchport general allowed vlan add 2 untagged
console(config-if)#
```

---

**Example 4** - Configures VLAN on port 21 as untagged on input and tagged on output:

---

```
console(config)# interface gi21
console(config-if)# switchport mode general
console(config-if)# switchport general pvid 2
console(config-if)# switchport general allowed vlan add 2 tagged
console(config-if)#
```

---

**Example 5** - Configures VLAN on port 14 as tagged on input and tagged on output:

---

```
console(config)# interface gi14
console(config-if)# switchport mode general
```

---

```
console(config-if)# switchport general allowed vlan add 2 tagged
console(config-if)#
```

---

**Example 6** - Configures VLAN on port 23 as tagged on input and untagged on output:

---

```
console(config)# interface gi23
console(config-if)# switchport mode general
console(config-if)# switchport general allowed vlan add 2 tagged
console(config-if)#
```

---

### 30.17 switchport general ingress-filtering disable

Use the **switchport general ingress-filtering disable** Interface Configuration (Ethernet, Port-channel) mode command to disable port ingress filtering (no packets are discarded at the ingress) on a general port. Use the no form of this command to restore the default configuration.

#### Syntax

**switchport general ingress-filtering disable**

**no switchport general ingress-filtering disable**

#### Parameters

N/A

#### Default Configuration

Ingress filtering is enabled.

#### Command Mode

Interface Configuration (Ethernet, port-channel) mode

#### Example

The following example disables port ingress filtering on `gi1`.

---

```
Console(config)# interface gi1
```

```
Console(config-if)# switchport mode general
Console(config-if)# switchport general ingress-filtering disable
```

---

## 30.18 switchport general acceptable-frame-type

The **switchport general acceptable-frame-type** Interface Configuration mode command configures the types of packets (tagged/untagged) that are filtered (discarded) on the interface. Use the **no** form of this command to return ingress filtering to the default.

### Syntax

```
switchport general acceptable-frame-type {tagged-only | untagged-only | all}
no switchport general acceptable-frame-type
```

### Parameters

- **tagged-only**—Ignore (discard) untagged packets and priority-tagged packets.
- **untagged-only**—Ignore (discard) VLAN-tagged packets (not including priority-tagged packets)
- **all**—Do not discard packets untagged or priority-tagged packets.

### Default Configuration

All frame types are accepted at ingress (**all**).

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### Example

The following example configures port `gi3` to be in general mode and to discard untagged frames at ingress.

---

```
Console(config)# interface gi3
Console(config-if)# switchport mode general
Console(config-if)# switchport general acceptable-frame-type tagged-only
```

---

### 30.19 switchport customer vlan

When a port is in customer mode it is in QinQ mode. This enables the user to use their own VLAN arrangements (PVID) across a provider network. The switch is in QinQ mode when it has one or more customer ports.

Use the **switchport customer vlan** Interface Configuration mode command to set the port's VLAN when the interface is in customer mode (set by [switchport mode](#)). Use the no form of this command to restore the default configuration.

#### Syntax

**switchport customer vlan** *vlan-id*

**no switchport customer vlan**

#### Parameters

**vlan** *vlan-id*—Specifies the customer VLAN.

#### Default Configuration

No VLAN is configured as customer.

#### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

#### Example

The following example defines `gi5` as a member of customer VLAN 5.

---

```
Console(config)# interface gi5
Console(config-if)# switchport mode customer
Console(config-if)# switchport customer vlan 5
```

---

### 30.20 map mac macs-group

Forwarding of packets based on their MAC address requires setting up groups of MAC addresses and then mapping these groups to VLANs.

Use the **map mac macs-group** VLAN Configuration mode command to map a MAC address or range of MAC addresses to a group of MAC addresses, which is then

used in `switchport general map macs-group vlan`. Use the **no** form of this command to delete the mapping.

This command can only be used when the device is in Layer 2 mode.

### Syntax

```
map mac mac-address [prefix-mask / host] macs-group group
```

```
no map mac mac-address [prefix-mask / host]
```

### Parameters

- **mac** *mac-address*—Specifies the MAC address to be mapped to the group of MAC addresses.
- **prefix-mask**—Specifies the number of ones in the mask.
- **host**—Specifies that the mask is comprised of all 1s.
- **macs-group** *group*—Specifies the group number (range: 1–2147483647)

### Default Configuration

N/A

### Command Mode

VLAN Configuration mode

### Example

The following example creates two groups of MAC addresses, sets a port to general mode and maps the groups of MAC addresses to specific VLANs.

```
Console(config)# vlan database
console(config-vlan)# map mac 0000.1111.0000 32 macs-group 1
console(config-vlan)# map mac 0000.0000.2222 host macs-group 2
console(config-vlan)# exit
console(config)# interface gill
console(config-if)# switchport mode general
console(config-if)# switchport general map macs-group 1 vlan 2
console(config-if)# switchport general map macs-group 2 vlan 3
```

---

## 30.21 switchport general map macs-group vlan

After groups of MAC addresses have been created (see [map mac macs-group](#)), they can be mapped to specific VLANs.

Use the **switchport general map macs-group vlan** Interface Configuration (Ethernet, Port-channel) mode command to set a MAC-based classification rule. Use the no form of this command to delete a classification rule.

### Syntax

```
switchport general map macs-group group vlan vlan-id
```

```
no switchport general map macs-group group
```

### Parameters

- **macs-group *group***—Specifies the group number (range: 1–2147483647)
- **vlan *vlan-id***—Defines the VLAN ID associated with the rule.

### Default Configuration

N/A

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### User Guidelines

MAC-based VLAN rules cannot contain overlapping ranges on the same interface.

The VLAN classification rule priorities are:

1. MAC-based VLAN (Best match among the rules).
2. Subnet-based VLAN (Best match among the rules).
3. Protocol-based VLAN.
4. PVID.

### Example

The following example creates two groups of MAC addresses, sets a port to general mode and maps the groups of MAC addresses to specific VLANs.

---

```
Console(config)# vlan database  
  
console(config-vlan)# map mac 0000.1111.0000 32 macs-group 1
```



```
console(config-vlan)# map mac 0000.0000.2222 host macs-group 2
console(config-vlan)# exit
console(config)# interface gill
console(config-if)# switchport mode general
console(config-if)# switchport general map macs-group 1 vlan 2
console(config-if)# switchport general map macs-group 2 vlan 3
```

---

## 30.22 show vlan macs-groups

Use the **show vlan macs-groups** EXEC mode command to display the MAC addresses that belong to the defined MACs-groups.

### Syntax

**show vlan macs-groups**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

EXEC mode

### Example

The following example displays macs-groups information.

---

```
console# show vlan macs-groups
```

MAC Address	Mask	Group ID
00:12:34:56:78:90	20	22
00:60:70:4c:73:ff	40	1

---

### 30.23 **switchport forbidden default-vlan**

Use the **switchport forbidden default-vlan** Interface Configuration command to forbid a port from being added to the default VLAN. Use the no form of this command to revert to default.

#### Syntax

**switchport forbidden default-vlan**

**no switchport forbidden default-vlan**

#### Parameters

N/A

#### Default Configuration

Membership in the default VLAN is allowed.

#### Command Mode

Interface and Interface range configuration (Ethernet, port-channel)

#### User Guidelines

The command may be used at any time regardless of whether the port belongs to the default VLAN.

The **no** command does not add the port to the default VLAN, it only defines an interface as permitted to be a member of the default VLAN, and the port will be added only when conditions are met.

#### Example

The following example forbids the port gi1 from being added to the default VLAN.

---

```
console(config)#interface gi1
console(config-if)# switchport forbidden default-vlan
```

---

### 30.24 **switchport forbidden vlan**

The **switchport forbidden vlan** Interface Configuration (Ethernet, Port-channel) mode command forbids adding or removing specific VLANs to or from a port. To restore the default configuration, use the **no** form of this command.

### Syntax

**switchport forbidden vlan** {**add** *vlan-list*| **remove** *vlan-list*}

**no switchport forbidden vlan** {**add** *vlan-list*| **remove** *vlan-list*}

### Parameters

- **add** *vlan-list* — Specifies a list of VLAN IDs to add. Separate nonconsecutive VLAN IDs with a comma and no spaces; use a hyphen designate a range of IDs.
- **remove** *vlan-list* — Specifies a list of VLAN IDs to remove. Separate nonconsecutive VLAN IDs with a comma and no spaces; use a hyphen designate a range of IDs.

### Default Configuration

All VLANs are allowed.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### Example

The following example forbids adding VLAN IDs 234 to 256 to `gi7`.

---

```
Console(config)# interface gi7
Console(config-if)# switchport mode trunk
Console(config-if)# switchport forbidden vlan add 234-256
```

---

---

## 30.25 switchport default-vlan tagged

Use the **switchport default-vlan tagged** Interface Configuration command to configure the port as a tagged port in the default VLAN. Use the **no** form of the command to return the port to an untagged port.

### Syntax

**switchport default-vlan tagged**

**no switchport default-vlan tagged****Parameters**

N/A

**Default Configuration**

If the port is a member in the default VLAN, by default, it is a member as an untagged port.

**Command Mode**

Interface configuration (Ethernet, port-channel)

**User Guidelines**

The command adds a port to the default VLAN as a tagged port.

The command is available only if the port mode is trunk or general.

When a trunk port is a member in the default VLAN as a tagged port then:

- The native VLAN cannot be the default VLAN
- The default of the native VLAN is 4095

Note: If the native VLAN of a port is the default VLAN when the port is added to the default VLAN as tagged, the native VLAN is set by the system to 4095.

When a general port is a member in the default VLAN as a tagged port then:

- The PVID can be the default VLAN.
- The default PVID is the default VLAN.

Note: The PVID is not changed when the port is added to the default VLAN as a tagged.

When executing the **switchport default-vlan tagged** command, the port is added (automatically by the system) to the default VLAN when the following conditions no longer exist:

- The port is a member in a LAG.
- The port is 802.1X unauthorized.
- An IP address is defined on the port.
- The port is a destination port of port mirroring.
- An IP address is defined on the default VLAN and the port is a PVE protected port.

The **no switchport default-vlan tagged** command removes the port from the default VLAN, and returns the default VLAN mode to untagged.

Note:

- If the native VLAN of a trunk port is 4095 when the port is removed from the default VLAN (as a tagged), the native VLAN is set by the system to the default VLAN.
- The PVID of a general port is not changed when the port is removed from the default VLAN (as a tagged). If the PVID is the default VLAN, the port is added by the system to the default VLAN as an untagged.

### Example

The following example configures the port gi1 as a tagged port in the default VLAN.

```
Console(config)#interface gi1
console(config-if)# switchport mode trunk
Console(config-if)#switchport default-vlan tagged
```

## 30.26 show interfaces switchport

Use the **show interfaces switchport** Privileged EXEC command to display the administrative and operational status of all interfaces or a specific interface.

### Syntax

```
show interfaces switchport [interface-id]
```

### Parameters

**interface-id**—Specify an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel

### Default Configuration

Displays information for all interfaces.

### Command Mode

EXEC mode

**Examples:**

**Example 1** - The following example displays the the command output for a trunk port:

---

```
Console> show interfaces switchport gi1

Port gi1:

Port Mode: Trunk
Gvrp Status: disabled
Ingress Filtering: true
Acceptable Frame Type: admitAll
Ingress UnTagged VLAN ( NATIVE ): 2
Protected: Enabled, Uplink is gi9.

Port gi1 is member in:

  VLAN   Name           Egress Rule  Type
  ----   -
  1      default        untagged     System
  8      VLAN008        tagged       Dynamic
  11     VLAN0011       tagged       Static
  19     IPv6VLAN       untagged     Static
  72     VLAN0072       untagged     Static

Forbidden VLANS:

  VLAN   Name
  ----   -
  73     Out

Classification rules:

Mac based VLANs:

  Group ID  Vlan ID
```

---

**Example 2** - The following example displays the output for a general port:

```
Console> show interfaces switchport gi2
```

```
Port gi2:
VLAN Membership mode: General
Operating Parameters:
PVID: 4095 (discard vlan)
Ingress Filtering: Enabled
Acceptable Frame Type: All
GVRP status: Enabled
Protected: Disabled

Port gi1 is member in:
VLAN  Name          Egress Rule Type
----  -
91    IP Telephony    tagged    Static

Protected: Disabled

Port gi2 is statically configured to:
VLAN  Name          Egress Rule Type
----  -
8     VLAN0072      untagged
91    IP Telephony    tagged
```

```
Forbidden VLANS:
VLAN  Name
----  -
73    Out
```

---

**Example 3** - The following example displays the command output for an access port:

```
Console> show interfaces switchport gi2
Port gi2:
```

```

Port Mode: Access

Gvrp Status: disabled

Ingress Filtering: true

Acceptable Frame Type: admitAll

Ingress UnTagged VLAN ( NATIVE ): 1

```

Port is member in:

Vlan	Name	Egress Rule	Port Membership Type
1	1	Untagged	System

Forbidden VLANS:

Vlan	Name
-----	-----

Classification rules:

Mac based VLANs:

## 30.27 ip internal-usage-vlan

The system assigns a VLAN to every IP address. In rare cases, this might conflict with a user requirement for that VLAN. In this case, use the **ip internal-usage-vlan** Interface Configuration (Ethernet, Port-channel) mode command to reserve a different VLAN as the internal usage VLAN of an interface. Use the **no** form of this command to restore the default configuration.

### Syntax

```
ip internal-usage-vlan vlan-id
```

```
no ip internal-usage-vlan
```

### Parameters

**vlan-id**—Specifies the internal usage VLAN ID.



### Default Configuration

No VLAN is reserved as an internal usage VLAN by default (using this command).

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode. It cannot be configured for a range of interfaces (range context).

### User Guidelines

An internal usage VLAN is assigned by the system when an IP interface is defined on an Ethernet port or port-channel.

If an internal usage VLAN is not defined for a port, the software selects one of the unused VLANs.

If a VLAN was chosen by the software for internal usage, but you want to use that VLAN for a static or dynamic VLAN, do one of the following:

- Remove the IP address from the interface (this releases the internal usage VLAN).
- Recreate the VLAN on the required interface (now it will be assigned to the interface and not be used as an internal usage VLAN)
- Recreate the IP interface (another internal usage VLAN is assigned to this IP interface) or use this command to explicitly define the internal usage VLAN.

### Example

The following example reserves unused VLAN 200 as the internal usage VLAN of gi3.

```
Console(config)# interface gi3
Console(config-if)# ip internal-usage-vlan 200
```

## 30.28 show vlan internal usage

Use the **show vlan internal usage** Privileged EXEC mode command to display a list of VLANs used internally by the device (defined by the user).

### Syntax

**show vlan internal usage**

**Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

Privileged EXEC mode

**Example**

The following example displays VLANs used internally by the device.

```
Console# show vlan internal usage
```

Usage	VLAN	Reserved	IP address
-----	-----	-----	-----
gi21	1007	No	Active
gi22	1008	Yes	Inactive
gi23	1009	Yes	Active

# 31 Internet Group Management Protocol (IGMP) Snooping Commands

---

## 31.1 ip igmp snooping (Global)

Use the **ip igmp snooping** Global Configuration mode command to enable Internet Group Management Protocol (IGMP) snooping. Use the **no** form of this command to disable IGMP snooping.

### Syntax

**ip igmp snooping**

**no ip igmp snooping**

### Default Configuration

Disabled.

### Command Mode

Global Configuration mode

### Example

The following example enables IGMP snooping.

---

```
Console(config)# ip igmp snooping
```

---

## 31.2 ip igmp snooping vlan

Use the **ip igmp snooping vlan** Global Configuration mode command to enable IGMP snooping on a specific VLAN. Use the **no** form of this command to disable IGMP snooping on a VLAN interface.

### Syntax

**ip igmp snooping vlan *vlan-id***

**no ip igmp snooping vlan *vlan-id***

**Parameters**

**vlan** *vlan-id*—Specifies the VLAN.

**Default Configuration**

Disabled

**Command Mode**

Global Configuration mode

**User Guidelines**

IGMP snooping can be enabled only on static VLANs.

IGMPv1, IGMPv2 and IGMPv3 are supported.

To activate IGMP snooping, the [bridge multicast filtering](#) should be enabled.

The user guidelines of the [bridge multicast mode](#) Interface VLAN Configuration command describes the configuration that is written into the FDB as a function of the FDB mode and the IGMP version that is used in the network.

**Example**

```
console(config)# ip igmp snooping vlan 2
```

---

**31.3 ip igmp snooping vlan mrouter**

Use the **ip igmp snooping mrouter** Global Configuration mode command to enable automatic learning of Multicast router ports on a VLAN. Use the **no** form of this command to remove the configuration.

**Syntax**

**ip igmp snooping vlan** *vlan-id* **mrouter learn pim-dvmrp**

**no ip igmp snooping vlan** *vlan-id* **mrouter learn pim-dvmrp**

**Parameters**

**vlan** *vlan-id*—Specifies the VLAN.

**Default Configuration**

**Learning pim-dvmrp** is enabled.

### Command Mode

Global Configuration mode

### User Guidelines

Multicast router ports are learned according to:

- Queries received on the port
- PIM/PIMv2 received on the port
- DVMRP received on the port
- MRDISC received on the port
- MOSPF received on the port

You can execute the command before the VLAN is created.

### Example

```
console(config)# ip igmp snooping vlan 1 mrouter learn pim-dvmrp
```

## 31.4 ip igmp snooping vlan mrouter interface

Use the **ip igmp snooping mrouter interface** Global Configuration mode command to define a port that is connected to a Multicast router port. Use the **no** form of this command to remove the configuration.

### Syntax

```
ip igmp snooping vlan vlan-id mrouter interface interface-list
```

```
no ip igmp snooping vlan vlan-id mrouter interface interface-list
```

### Parameters

- **vlan** *vlan-id*—Specifies the VLAN.
- **interface** *interface-list*—Specifies the list of interfaces. The interfaces can be one of the following types: Ethernet port or Port-channel.

### Default Configuration

No ports defined

### Command Mode

Global Configuration mode

### User Guidelines

A port that is defined as a Multicast router port receives all IGMP packets (reports and queries) as well as all Multicast data.

You can execute the command before the VLAN is created.

### Example

```
console(config)# ip igmp snooping vlan 1 mrouter interface gil/1/1
```

---

## 31.5 ip igmp snooping vlan forbidden mrouter interface

Use the **ip igmp snooping forbidden mrouter interface** Global Configuration mode command to forbid a port from being defined as a Multicast router port by static configuration or by automatic learning. Use the **no** form of this command to remove the configuration.

### Syntax

**ip igmp snooping vlan** *vlan-id* **forbidden mrouter interface** *interface-list*

**no ip igmp snooping vlan** *vlan-id* **forbidden mrouter interface** *interface-list*

### Parameters

- **vlan** *vlan-id*—Specifies the VLAN.
- **interface** *interface-list*—Specifies a list of interfaces. The interfaces can be from one of the following types: Ethernet port or Port-channel.

### Default Configuration

No ports defined.

### Command Mode

Global Configuration mode

### User Guidelines

A port that is a forbidden mrouter port cannot be a Multicast router port (i.e. cannot be learned dynamically or assigned statically).

You can execute the command before the VLAN is created.

### Example

```
console(config)# ip igmp snooping vlan 1 forbidden mrouter interface gil/1/1
```

## 31.6 ip igmp snooping vlan static

Use the **ip igmp snooping static** Global Configuration mode command to register an IP-layer Multicast address to the bridge table, and to add static ports to the group. Use the **no** form of this command to remove ports specified as members of a static Multicast group.

### Syntax

```
ip igmp snooping vlan vlan-id static ip-address [interface interface-list]
```

```
no ip igmp snooping vlan vlan-id static ip-address [interface interface-list]
```

### Parameter

- **vlan** *vlan-id*—Specifies the VLAN.
- **static** *ip-address*—Specifies the IP Multicast address.
- **interface** *interface-list*—Specifies a list of interfaces. The interfaces can be from one of the following types: Ethernet port or Port-channel.

### Default Configuration

No Multicast addresses are defined.

### Command Mode

Global Configuration mode

### User Guidelines

Static Multicast addresses can only be defined on static VLANs.

You can execute the command before the VLAN is created.

You can register an entry without specifying an interface.

Using the **no** command without a port-list removes the entry.

### Example

```
console(config)# ip igmp snooping vlan 1 static 239.2.2.2 interface gil/1/1
```

## 31.7 ip igmp snooping vlan querier

Use the **ip igmp snooping vlan querier** Global Configuration mode command to enable the Internet Group Management Protocol (IGMP) querier on a specific VLAN. Use the **no** form of this command to disable the IGMP querier on a VLAN interface.

### Syntax

```
ip igmp snooping vlan vlan-id querier
```

```
no ip igmp snooping vlan vlan-id querier
```

### Parameters

**vlan** *vlan-id*—Specifies the VLAN

### Default Configuration

Disabled

### Command Mode

Global Configuration mode

### User Guidelines

The IGMP snooping querier can be enabled on a VLAN only if IGMP snooping is enabled for that VLAN.

At most one switch can be configured as an IGMP Querier for a VLAN.

When the IGMP snooping querier is enabled, it starts after a host-time-out/2 with no IGMP traffic being detected from a Multicast router.

The IGMP Snooping Querier disables itself if it detects IGMP traffic from a Multicast router. It restarts automatically after host-time-out/2.



Following are the IGMP snooping querier parameters as a function of the IGMP snooping parameters:

- QueryMaxResponseTime: host-time-out/10.
- QueryInterval: host-time-out/ 3.

### Example

```
console(config)# ip igmp snooping vlan 1 querier
```

## 31.8 ip igmp snooping vlan querier address

Use the **ip igmp snooping vlan querier address** Global Configuration mode command to define the source IP address that the IGMP snooping querier uses. Use the **no** form of this command to return to default.

### Syntax

```
ip igmp snooping vlan vlan-id querier address ip-address
```

```
no ip igmp snooping vlan vlan-id querier address
```

### Parameters

- **vlan *vlan-id***—Specifies the VLAN.
- **querier address *ip-address***—Source IP address.

### Default Configuration

If an IP address is configured for the VLAN, it is used as the source address of the IGMP snooping querier. If there are multiple IP addresses, the minimum IP address defined on the VLAN is used.

### Command Mode

Global Configuration mode

### User Guidelines

If an IP address is not configured by this command, and no IP address is configured for the querier's VLAN, the querier is disabled.

---

**Example**

```
console(config)# ip igmp snooping vlan 1 querier address 10.5.234.205
```

---

**31.9 ip igmp snooping vlan querier version**

Use the **ip igmp snooping vlan querier version** Global Configuration mode command to configure the IGMP version of an IGMP querier on a specific VLAN. Use the **no** form of this command to return to the default version.

**Syntax**

```
ip igmp snooping vlan vlan-id querier version {2|3}
```

```
no ip igmp snooping vlan vlan-id querier version
```

**Parameters**

- **vlan *vlan-id***—Specifies the VLAN.
- **querier version 2**—Specifies that the IGMP version would be IGMPv2.
- **querier version 3**—Specifies that the IGMP version would be IGMPv3.

**Default Configuration**

IGMPv2.

**Command Mode**

Global Configuration mode

**Example**

---

```
console(config)# ip igmp snooping vlan 1 querier version 3
```

---

**31.10 ip igmp robustness**

Use the **ip igmp robustness** Interface Configuration (VLAN) mode command to set the IGMP robustness variable on a VLAN. Use the **no** format of the command to return to default.

**Syntax**

**ip igmp robustness** *count*

**no ip igmp robustness**

**Parameters**

**count**—The number of expected packet loss on a link. Parameter range. (Range: 1–7)

**Default Configuration**

2

**Command Mode**

Interface Configuration (VLAN) mode

**User Guidelines**

You can execute the command before the VLAN is created, but you must enter the command in Interface VLAN mode.

**Example**

---

```
console(config)# interface vlan 1
console(config-if)# ip igmp robustness 3
```

---

**31.11 ip igmp query-interval**

Use the **ip igmp query-interval** Interface Configuration (VLAN) mode command to configure the Query interval on a VLAN. Use the **no** format of the command to return to default.

**Syntax**

**ip igmp query-interval** *seconds*

**no ip igmp query-interval**

**Parameters**

**seconds**—Frequency, in seconds, at which IGMP query messages are sent on the interface. (Range: 30–18000)

**Default Configuration**

125

**Command Mode**

Interface Configuration (VLAN) mode

**User Guidelines**

You can execute the command before the VLAN is created.

**Example**

```
console(config)# interface vlan 1
console(config-if)# ip igmp query-interval 200
```

---

**31.12 ip igmp query-max-response-time**

Use the **ip igmp query-max-response-time** Interface Configuration (VLAN) mode command to configure the Query Maximum Response time on a VLAN. Use the **no** format of the command to return to default.

**Syntax**

**ip igmp query-max-response-time** *seconds*

**no ip igmp query-max-response-time**

**Parameters**

**seconds**—Maximum response time, in seconds, advertised in IGMP queries. (Range: 5–20)

**Default Configuration**

10

**Command Mode**

Interface Configuration (VLAN) mode

**User Guidelines**

You can execute the command before the VLAN is created.

---

### Example

```
console(config)# interface vlan 1
console(config-if)# ip igmp query-max-response-time 20
```

---

## 31.13 ip igmp last-member-query-count

Use the **ip igmp last-member-query-count** Interface Configuration (VLAN) mode command to configure the Last Member Query Counter on a VLAN. Use the **no** format of the command to return to default.

### Syntax

**ip igmp last-member-query-count** *count*

**no ip igmp last-member-query-count**

### Parameter

**count**—The number of times that group- or group-source-specific queries are sent upon receipt of a message indicating a leave. (Range: 1–7)

### Default Configuration

A value of Robustness variable

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

You can execute the command before the VLAN is created.

### Example

---

```
console(config)# interface vlan 1
console(config-if)# ip igmp last-member-query-count 7
```

---

### 31.14 ip igmp last-member-query-interval

Use the **ip igmp last-member-query-interval** Interface Configuration (VLAN) mode command to configure the Last Member Query interval on a VLAN. Use the **no** format of the command to return to default.

#### Syntax

**ip igmp last-member-query-interval** *milliseconds*

**no ip igmp last-member-query-interval**

#### Parameters

**milliseconds**—Interval, in milliseconds, at which IGMP group-specific host query messages are sent on the interface. (Range: 100–25500)

#### Default Configuration

1000

#### Command Mode

Interface Configuration (VLAN) mode

#### User Guidelines

You can execute the command before the VLAN is created.

#### Example

---

```
console(config)# interface vlan 1
console(config-if)# ip igmp last-member-query-interval 2000
```

---

### 31.15 ip igmp snooping vlan immediate-leave

Use the **ip igmp snooping vlan immediate-leave** Global Configuration mode command to enable the IGMP Snooping Immediate-Leave processing on a VLAN. Use the **no** format of the command to disable IGMP Snooping Immediate-Leave processing.

#### Syntax

**ip igmp snooping vlan** *vlan-id* **immediate-leave**

**no ip igmp snooping vlan *vlan-id* immediate-leave****Parameters**

**vlan *vlan-id***—Specifies the VLAN ID value. (Range: 1–4094)

**Default Configuration**

Disabled

**Command Mode**

Global Configuration mode

**User Guidelines**

You can execute the command before the VLAN is created.

**Example**

The following example enables IGMP snooping immediate-leave feature on VLAN 1.

---

```
Console(config)# ip igmp snooping vlan 1 immediate-leave
```

---

**31.16 show ip igmp snooping mrouter**

The **show ip igmp snooping mrouter** EXEC mode command displays information on dynamically learned Multicast router interfaces for all VLANs or for a specific VLAN.

**Syntax**

**show ip igmp snooping mrouter** [*interface *vlan-id**]

**Parameters**

**interface *vlan-id***—Specifies the VLAN ID.

**Command Mode**

EXEC mode

**Example**

The following example displays information on dynamically learned Multicast router interfaces for VLAN 1000.

---

```
Console# show ip igmp snooping mrouter interface 1000
```

VLAN	Dynamic	Static	Forbidden
----	-----	-----	-----
1000	gil/1/1	gil/1/2	gil/1/3-23

**31.17 show ip igmp snooping interface**

The **show ip igmp snooping interface** EXEC mode command displays the IGMP snooping configuration for a specific VLAN.

**Syntax**

```
show ip igmp snooping interface vlan-id
```

**Parameters**

**interface** *vlan-id*—Specifies the VLAN ID.

**Command Mode**

EXEC mode

**Example**

The following example displays the IGMP snooping configuration for VLAN 1000

---

```
Console # show ip igmp snooping interface 1000
IGMP Snooping is globally enabled
IGMP Snooping admin: Enabled
IGMP Snooping oper: Enabled
Routers IGMP version: 3
Groups that are in IGMP version 2 compatibility mode:
231.2.2.3, 231.2.2.3
```



Groups that are in IGMP version 1 compatibility mode:

```
IGMP snooping querier admin: Enabled
```

```
IGMP snooping querier oper: Enabled
```

```
IGMP snooping querier address admin:
```

```
IGMP snooping querier address oper: 172.16.1.1
```

```
IGMP snooping querier version admin: 3
```

```
IGMP snooping robustness: admin 2 oper 2
```

```
IGMP snooping query interval: admin 125 sec oper 125 sec
```

```
IGMP snooping query maximum response: admin 10 sec oper 10 sec
```

```
IGMP snooping last member query counter: admin 2 oper 2
```

```
IGMP snooping last member query interval: admin 1000 msec oper 500 msec
```

```
IGMP snooping last immediate leave: enable
```

```
Automatic learning of Multicast router ports is enabled
```

---

## 31.18 show ip igmp snooping groups

The **show ip igmp snooping groups** EXEC mode command displays the Multicast groups learned by the IGMP snooping.

### Syntax

```
show ip igmp snooping groups [vlan vlan-id] [address ip-multicast-address]  
[source ip-address]
```

### Parameters

**vlan** *vlan-id*—Specifies the VLAN ID.

**address** *ip-multicast-address*—Specifies the IP multicast address.

**source** *ip-address*—Specifies the IP source address.

**Command Mode**

EXEC mode

**User Guidelines**

To see all Multicast groups learned by IGMP snooping, use the **show ip igmp snooping groups** command without parameters.

Use the **show ip igmp snooping groups** command with parameters to see a needed subset of all Multicast groups learned by IGMP snooping

To see the full Multicast address table (including static addresses), use the **show bridge multicast address-table** command.

**Example**

The following example shows sample output for IGMP version 2.

---

```
Console# show ip igmp snooping groups
```

Vlan	Group Address	Source Address	Include Ports	Exclude Ports	Comp-Mod e
----	-----	-----	-----	-----	-----
1	239.255.255.2	*	gi1		-----
	50				v3

---

## 32 IPv6 MLD Snooping Commands

---

### 32.1 ipv6 mld snooping (Global)

The **ipv6 mld snooping** Global Configuration mode command enables IPv6 Multicast Listener Discovery (MLD) snooping. To disable IPv6 MLD snooping, use the **no** form of this command.

#### Syntax

**ipv6 mld snooping**

**no ipv6 mld snooping**

#### Default Configuration

IPv6 MLD snooping is disabled.

#### Command Mode

Global Configuration mode

#### Example

The following example enables IPv6 MLD snooping.

---

```
Console(config)# ipv6 mld snooping
```

---

### 32.2 ipv6 mld snooping vlan

Use the **ipv6 mld snooping vlan** Global Configuration mode command to enable MLD snooping on a specific VLAN. Use the **no** form of this command to disable MLD snooping on a VLAN interface.

#### Syntax

**ipv6 mld snooping vlan** *vlan-id*

**no ipv6 mld snooping vlan** *vlan-id*

#### Parameters

*vlan-id*—Specifies the VLAN.

### Default Configuration

Disabled

### Command Mode

Global Configuration mode

### User Guidelines

MLD snooping can only be enabled on static VLANs.

MLDv1 and MLDv2 are supported.

To activate MLD snooping, the Bridge Multicast Filtering command should be enabled.

The user guidelines of the bridge multicast IPv6 mode interface VLAN configuration command describe the configuration that can be written into the FDB as a function of the FDB mode, and the MLD version that is used in the network.

### Example

```
console(config)# ipv6 mld snooping vlan 2
```

---

## 32.3 ipv6 mld robustness

Use the **ipv6 mld robustness** interface Configuration mode command to change a value of the MLD robustness variable. Use the **no** format of the command to return to default.

### Syntax

```
ipv6 mld robustness count
```

```
no ipv6 mld robustness
```

### Parameters

**count** - The number of expected packet losses on a link. (Range: 1–7)

### Default Configuration

2

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

You can execute the command before the VLAN is created.

### Example

```
console(config)# interface vlan 1
console(config-if)# ipv6 mld robustness 3
```

## 32.4 ipv6 mld snooping mrouter

Use the **ipv6 mld snooping mrouter** Global Configuration mode command to enable automatic learning of multicast router ports. Use the **no** form of this command to remove the configuration.

### Syntax

```
ipv6 mld snooping vlan vlan-id mrouter learn pim-dvmrp
no ipv6 mld snooping vlan vlan-id mrouter learn pim-dvmrp
```

### Parameters

**vlan-id**—Specifies the VLAN.

### Default Configuration

Learning **pim-dvmrp** is enabled.

### Command Mode

Global Configuration mode

### User Guidelines

Multicast router ports can be configured statically with the **bridge multicast forward-all** command.

You can execute the command before the VLAN is created.

---

### Example

```
console(config)# ipv6 mld snooping vlan 1 mrouter learn pim-dvmrp
```

---

## 32.5 ipv6 mld snooping mrouter interface

Use the **ipv6 mld snooping mrouter interface** Global Configuration mode command to define a port that is connected to a multicast router port. Use the **no** form of this command to remove the configuration.

### Syntax

```
ipv6 mld snooping vlan vlan-id mrouter interface interface-list
```

```
no ipv6 mld snooping vlan vlan-id mrouter interface interface-list
```

### Parameters

- **vlan-id**—Specifies the VLAN.
- **interface-list**—Specifies a list of interfaces. The interfaces can be from one of the following types: port or port-channel.

### Default Configuration

No ports defined

### Command Mode

Global Configuration mode

### User Guidelines

This command may be used in conjunction with the **bridge multicast forward-all** command, which is used in older versions to statically configure a port as a multicast router.

A port that is defined as a multicast router port receives all MLD packets (reports and queries) as well as all multicast data.

You can execute the command before the VLAN is created.

### Example

---

```
console(config)# ipv6 mld snooping vlan 1 mrouter interface gil
```

---

## 32.6 ipv6 mld snooping forbidden mrouter interface

Use the **ipv6 mld snooping forbidden mrouter interface** Global Configuration mode command to forbid a port from being defined as a multicast router port by static configuration or by automatic learning. Use the **no** form of this command to remove the configuration.

### Syntax

**ipv6 mld snooping** *vlan* *vlan-id* **forbidden mrouter** *interface* *interface-list*

**no ipv6 mld snooping** *vlan* *vlan-id* **forbidden mrouter** *interface* *interface-list*

### Parameters

- **vlan-id**—Specifies the VLAN.
- **interface-list**—Specifies list of interfaces. The interfaces can be from one of the following types: Ethernet port or Port-channel.

### Default Configuration

No forbidden ports by default

### Command Mode

Global Configuration mode

### User Guidelines

A port that is forbidden mrouter port cannot be a multicast router port (i.e. cannot be learned dynamically or assigned statically).

The command bridge **multicast forbidden forward-all** command was used in older versions to forbid dynamic learning of multicast router ports.

You can execute the command before the VLAN is created.

### Example

---

```
console(config)# ipv6 mld snooping vlan 1 forbidden mrouter interface gil
```

---

## 32.7 ipv6 mld snooping static

Use the **ipv6 mld snooping static** Global Configuration mode command to register a IPv6-layer multicast address to the bridge table, and to add statically ports to the

group. Use the **no** form of this command to remove ports specified as members of a static multicast group.

### Syntax

**ipv6 mld snooping** *vlan* *vlan-id* **static** *ipv6-address* **interface** [*interface-list*]

**no ipv6 mld snooping** *vlan* *vlan-id* **static** *ipv6-address* **interface** [*interface-list*]

### Parameters

- **vlan-id**—Specifies the VLAN.
- **ipv6-address**—Specifies the IP multicast address
- **interface-list**—Specifies list of interfaces. The interfaces can be from one of the following types: Ethernet port or Port-channel.

### Default Configuration

No multicast addresses are defined.

### Command Mode

Global configuration mode

### User Guidelines

Static multicast addresses can only be defined on static VLANs.

You can execute the command before the VLAN is created.

You can register an entry without specifying an interface.

Using the **no** command without a port-list removes the entry.

### Example

```
console(config)# ipv6 mld snooping vlan 1 static 239.2.2.2 gil
```

---

## 32.8 ipv6 mld query-interval

Use the **ipv6 mld query-interval** Interface Configuration mode command to configure the Query interval. Use the **no** format of the command to return to default.



### Syntax

**ipv6 mld query-interval** *seconds*

**ipv6 mld query-interval**

### Parameters

**seconds**—Frequency, in seconds, at which MLD query messages are sent on the interface. (Range: 30–18000)

### Default Configuration

125

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

You can execute the command before the VLAN is created.

### Example

```
console(config)# interface vlan 1
console(config-if)# ipv6 mld query-interval 3000
```

---

## 32.9 ipv6 mld query-max-response-time

Use the **ipv6 mld query-max-response-time** Interface Configuration mode command to configure the Query Maximum Response time. Use the **no** format of the command to return to default.

### Syntax

**ipv6 mld query-max-response-time** *seconds*

**no ipv6 mld query-max-response-time**

### Parameter

**seconds**—Maximum response time, in seconds, advertised in MLD queries. (Range: 5–20)

**Default Configuration**

10

**Command Mode**

Interface Configuration (VLAN) mode

**User Guidelines**

You can execute the command before the VLAN is created.

**Example**

---

```
console(config)# interface vlan 1
console(config-if)# ipv6 mld query-max-response-time 5
```

---

**32.10 ipv6 mld last-member-query-count**

Use the **ipv6 mld last-member-query-count** Interface Configuration mode command to configure the Last Member Query Counter. Use the **no** format of the command to return to default.

**Syntax**

```
ipv6 mld last-member-query-count count
no ipv6 mld last-member-query-count
```

**Parameters**

**count**—The number of times that group- or group-source-specific queries are sent upon receipt of message indicating a leave. (Range: 1–7)

**Default Configuration**

A value of Robustness variable

**Command Mode**

Interface Configuration (VLAN) mode

**User Guidelines**

You can execute the command before the VLAN is created.

---

### Example

```
console(config)# interface vlan 1
console(config-if)# ipv6 mld last-member-query-count 3
```

---

## 32.11 ipv6 mld last-member-query-interval

Use the **ipv6 mld last-member-query-interval** interface configuration command to configure the Last Member Query Interval. Use the **no** format of the command to return to default.

### Syntax

**ipv6 mld last-member-query-interval** *milliseconds*

**no ipv6 mld last-member-query-interval**

### Parameter

**milliseconds**—Interval, in milliseconds, at which MLD group-specific host query messages are sent on the interface. (Range: 100–64512).

### Default Configuration

1000

### Command Mode

Interface Configuration (VLAN) mode

### User Guidelines

You can execute the command before the VLAN is created.

### Example

```
console(config)# interface vlan 1
console(config-if)# ipv6 mld last-member-query-interval 2000
```

---

## 32.12 ipv6 mld snooping vlan immediate-leave

Use the **ipv6 mld snooping vlan immediate-leave** Global Configuration mode command to enable MLD Snooping Immediate-Leave processing on a VLAN. Use

the **no** format of the command to return to disable MLD Snooping Immediate-Leave processing.

### Syntax

**ipv6 mld snooping vlan *vlan-id* immediate-leave**

**no ipv6 mld snooping vlan *vlan-id* immediate-leave**

### Parameters

**vlan-id**—Specifies the VLAN ID value. (Range: 1–4094)

### Default Configuration

Disabled

### Command Mode

Global Configuration mode

### User Guidelines

You can execute the command before the VLAN is created.

### Example

```
console(config)# ipv6 mld snooping vlan 1 immediate-leave
```

---

## 32.13 show ipv6 mld snooping mrouter

The **show ipv6 mld snooping mrouter** EXEC mode command displays information on dynamically learned multicast router interfaces for all VLANs or for a specific VLAN.

### Syntax

**show ipv6 mld snooping mrouter [*interface vlan-id*]**

### Parameters

**interface vlan-id**—Specifies the VLAN ID.

### Command Mode

EXEC mode

**Example**

The following example displays information on dynamically learned multicast router interfaces for VLAN 1000

---

```
Console# show ipv6 mld snooping mrouter interface 1000

VLAN   Static   Dynamic   Forbidden
----   -
1000   gi1      gi2       gi3-23
```

---

**32.14 show ipv6 mld snooping interface**

The **show ipv6 mld snooping interface** EXEC mode command displays the IPv6 MLD snooping configuration for a specific VLAN.

**Syntax**

**show ipv6 mld snooping interface** *vlan-id*

**Parameters**

**vlan-id**—Specifies the VLAN ID.

**Command Mode**

EXEC mode

**Example**

The following example displays the MLD snooping configuration for VLAN 1000.

---

```
Console# show ipv6 mld snooping interface 1000

MLD Snooping is globally enabled

MLD Snooping admin: Enabled

MLD snooping oper mode: Enabled

Routers MLD version: 2
```

---

```

Groups that are in MLD version 1 compatibility mode:

FF12::3, FF12::8

MLD snooping robustness:admin 2 oper 2
MLD snooping query interval: admin 125 sec oper 125 sec
MLD snooping query maximum response: admin 10 sec oper 10 sec
MLD snooping last member query counter: admin 2 oper 2
MLD snooping last member query interval: admin 1000 msec oper 600 msec
MLD snooping last immediate leave: enable
Automatic learning of multicast router ports is enabled

```

---

## 32.15 show ipv6 mld snooping groups

The **show ipv6 mld snooping groups** EXEC mode command displays the multicast groups learned by the MLD snooping.

### Syntax

```
show ipv6 mld snooping groups [vlan vlan-id] [address ipv6-multicast-address]
[source ipv6-address]
```

### Parameters

- **vlan vlan-id**—Specifies the VLAN ID.
- **address ipv6-multicast-address**—Specifies the IPv6 multicast address.
- **source ipv6-address**—Specifies the IPv6 source address.

### Command Mode

EXEC mode

### User Guidelines

To see the full multicast address table (including static addresses), use the **show bridge multicast address-table** command.

The Include list contains the ports which are in a forwarding state for this group according to the snooping database. In general, the Exclude list contains the ports which have issued an explicit Exclude for that specific source in a multicast group.

The Reporters That Are Forbidden Statically list contains the list of ports which have asked to receive a multicast flow but were defined as forbidden for that multicast group in a multicast bridge.

Note: Under certain circumstances, the Exclude list may not contain accurate information; for example, in the case when two Exclude reports were received on the same port for the same group but for different sources, the port will not be in the Exclude list but rather in the Include list

### Example

The following example shows the output for IPv6 MLD version 2.

```

Console# show ipv6 mld snooping groups

VLAN   Group      Source Address      Include Ports  Exclude      Compatibility Mode
----   -
1      FF12::3    FE80::201:C9FF:FE40:8001  gi1           -----      1
1      FF12::3    FE80::201:C9FF:FE40:8002  gi2           -----      1
19     FF12::3    FE80::201:C9FF:FE40:8003  gi9           -----      2
19     FF12::8    FE80::201:C9FF:FE40:8004  gi1           -----      2
19     FF12::8    FE80::201:C9FF:FE40:8005  gi10-11      gi2           -----      2
      FF12::8                                gi3           -----

MLD Reporters that are forbidden statically:
VLAN   Group      Source Address      Ports
----   -
1      FF12::3    FE80::201:C9FF:FE40:8001  gi8
19     FF12::3    FE80::201:C9FF:FE40:8001  gi9
      FF12::8

```

---

## 33 Link Aggregation Control Protocol (LACP) Commands

---

### 33.1 lacp system-priority

Use the **lacp system-priority** Global Configuration mode command to set the system priority. Use the **no** form of this command to restore the default configuration.

#### Syntax

**lacp system-priority** *value*

**no lacp system-priority**

#### Parameters

**value**—Specifies the system priority value. (Range: 1–65535)

#### Default Configuration

The default system priority is 1.

#### Command Mode

Global Configuration mode

#### Example

The following example sets the system priority to 120.

---

```
Console(config)# lacp system-priority 120
```

---

### 33.2 lacp port-priority

Use the **lacp port-priority** Interface Configuration (Ethernet) mode command to set the physical port priority. Use the **no** form of this command to restore the default configuration.

#### Syntax

**lacp port-priority** *value*

**no lacp port-priority**



### Parameters

**value**—Specifies the port priority. (Range: 1use the **no** form of this command65535)

### Default Configuration

The default port priority is 1.

### Command Mode

Interface Configuration (Ethernet) mode

### Example

The following example sets the priority of `gi6`.

---

```
console(config)# interface gi6
console(config-if)# lacp port-priority 247
```

---

## 33.3 lacp timeout

Use the **lacp timeout** Interface Configuration (Ethernet) mode command to assign an administrative LACP timeout to an interface. Use the **no** form of this command to restore the default configuration.

### Syntax

**lacp timeout** *{long / short}*

**no lacp timeout**

### Parameters

- **long**—Specifies the long timeout value.
- **short**—Specifies the short timeout value.

### Default Configuration

The default port timeout value is Long.

### Command Mode

Interface Configuration (Ethernet) mode

### Example

The following example assigns a long administrative LACP timeout to `gi6`.

---

```
Console(config)# interface gi6
Console(config-if)# lacp timeout long
```

---

## 33.4 show lacp

Use the **show lacp** EXEC mode command to display LACP information for all Ethernet ports or for a specific Ethernet port.

### Syntax

```
show lacp interface-id [parameters / statistics / protocol-state]
```

### Parameters

- **interface-id** —Specify an interface ID. The interface ID must be an Ethernet port
- **parameters**—Displays parameters only.
- **statistics**—Displays statistics only.
- **protocol-state**—Displays protocol state only.

### Command Mode

EXEC mode

### Example

The following example displays LACP information for `gi1`.

---

```
Console> show lacp ethernet gi1

Port gi1 LACP parameters:
  Actor
```

```
system priority:          1
system mac addr:         00:00:12:34:56:78
port Admin key:          30
port Oper key:           30
port Oper number:        21
port Admin priority:     1
port Oper priority:      1
port Admin timeout:      LONG
port Oper timeout:       LONG
LACP Activity:           ACTIVE
Aggregation:             AGGREGATABLE
synchronization:        FALSE
collecting:              FALSE
distributing:            FALSE
expired:                 FALSE
Partner
```

```

system priority:          0
system mac addr:        00:00:00:00:00:00
port Admin key:         0
port Oper key:          0
port Oper number:       0
port Admin priority:    0
port Oper priority:     0
port Admin timeout:     LONG
port Oper timeout:      LONG
LACP Activity:          PASSIVE
Aggregation:            AGGREGATABLE
synchronization:       FALSE
collecting:             FALSE
distributing:           FALSE
expired:                FALSE

Port gil LACP Statistics:
LACP PDUs sent:         2
LACP PDUs received:    2

Port gil LACP Protocol State:
  LACP State Machines:
    Receive FSM:        Port Disabled State
    Mux FSM:            Detached State
  Control Variables:

```

```

        BEGIN:                FALSE
        LACP_Enabled:         TRUE
        Ready_N:              FALSE
        Selected:              UNSELECTED
        Port_moved:           FALSE
        NNT:                   FALSE
        Port_enabled:         FALSE
    Timer counters:
        periodic tx timer:    0
        current while timer:  0
        wait while timer:     0

```

---

### 33.5 show lacp port-channel

Use the **show lacp port-channel** EXEC mode command to display LACP information for a port-channel.

#### Syntax

```
show lacp port-channel [port_channel_number]
```

#### Parameters

**port\_channel\_number**—Specifies the port-channel number.

#### Command Mode

EXEC mode

#### Example

The following example displays LACP information about port-channel 1.

---

```

Console> show lacp port-channel 1

Port-Channel 1:Port Type 1000 Ethernet
    Actor

```

```
System 1
Priority: 000285:0E1C00
MAC Address: 29
Admin Key: 29
Oper Key:

Partner
System 0
Priority: 00:00:00:00:00:00
MAC Address: 14
Oper Key:
```

---

## 34 GARP VLAN Registration Protocol (GVRP) Commands

---

### 34.1 gvrp enable (Global)

Use the **gvrp enable** Global Configuration mode command to enable the Generic Attribute Registration Protocol (GARP) VLAN Registration Protocol (GVRP) globally. Use the **no** form of this command to disable GVRP on the device.

#### Syntax

**gvrp enable**

**no gvrp enable**

#### Parameters

N/A

#### Default Configuration

GVRP is globally disabled.

#### Command Mode

Global Configuration mode

#### Example

The following example enables GVRP globally on the device.

---

```
Console(config)# gvrp enable
```

---

### 34.2 gvrp enable (Interface)

Use the **gvrp enable** Interface Configuration (Ethernet, Port-channel) mode command to enable GVRP on an interface. Use the **no** form of this command to disable GVRP on an interface.

#### Syntax

**gvrp enable**

**no gvrp enable**

### Default Configuration

GVRP is disabled on all interfaces.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

An access port does not dynamically join a VLAN because it is always a member of a single VLAN only. Membership in an untagged VLAN is propagated in the same way as in a tagged VLAN. That is, the PVID must be manually defined as the untagged VLAN ID.

### Example

The following example enables GVRP on `gi6`.

---

```
Console(config)# interface gi6
Console(config-if)# gvrp enable
```

---

## 34.3 `gvrp vlan-creation-forbid`

Use the `gvrp vlan-creation-forbid` Interface Configuration mode command to disable dynamic VLAN creation or modification. Use the `no` form of this command to enable dynamic VLAN creation or modification.

### Syntax

`gvrp vlan-creation-forbid`

`no gvrp vlan-creation-forbid`

### Default Configuration

Enabled.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode



### Example

The following example disables dynamic VLAN creation on `gi3`.

---

```
Console(config)# interface gi3
Console(config-if)# gvrp vlan-creation-forbid
```

---

## 34.4 gvrp registration-forbid

Use the **gvrp registration-forbid** Interface Configuration mode command to deregister all dynamic VLANs on a port and prevent VLAN creation or registration on the port. Use the **no** form of this command to allow dynamic registration of VLANs on a port.

### Syntax

**gvrp registration-forbid**

**no gvrp registration-forbid**

### Default Configuration

Dynamic registration of VLANs on the port is allowed.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### Example

The following example forbids dynamic registration of VLANs on `gi2`.

---

```
Console(config)# interface gi2
Console(config-if)# gvrp registration-forbid
```

---

## 34.5 clear gvrp statistics

Use the **clear gvrp statistics** Privileged EXEC mode command to clear GVRP statistical information for all interfaces or for a specific interface.

**Syntax**

**clear gvrp statistics** [*interface-id*]

**Parameters**

**Interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Default Configuration**

All GVRP statistics are cleared.

**Command Mode**

Privileged EXEC mode

**Example**

The following example clears all GVRP statistical information on gi5.

---

```
Console# clear gvrp statistics gi5
```

---

## 34.6 show gvrp configuration

Use the **show gvrp configuration** EXEC mode command to display GVRP configuration information, including timer values, whether GVRP and dynamic VLAN creation are enabled, and which ports are running GVRP.

**Syntax**

**show gvrp configuration** [*interface-id*]

**Parameters**

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Default Configuration**

All GVRP statistics are displayed.

**Command Mode**

EXEC mode

**Example**

The following example displays GVRP configuration.

```

console# show gvrp configuration

GVRP Feature is currently Enabled on the device.

Maximum VLANs: 4094

Port  GVRP-Status  Regist-   Dynamic      Timers(ms)
      ation        ration   VLAN Creation  Leave  Join  Leave All
-----
gi1   Enabled        Forbidden Disabled      200   600  10000
gi2   Enabled        Normal   Enabled      400  1200  20000

```

**34.7 show gvrp statistics**

Use the **show gvrp statistics** EXEC mode command to display GVRP statistics for all interfaces or for a specific interface.

**Syntax**

```
show gvrp statistics [interface-id]
```

**Parameters**

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Default Configuration**

All GVRP statistics are displayed.

**Command Mode**

EXEC mode

**Example**

The following example displays GVRP statistical information.

```

Console# show gvrp statistics

GVRP statistics:
-----

Legend:
rJE :   Join Empty Received      rJIn: Join In Received
rEmp:   Empty Received           rLIn: Leave In Received
rLE :   Leave Empty Received     rLA : Leave All Received
sJE :   Join Empty Sent          sJIn: Join In Sent
sEmp:   Empty Sent              sLIn: Leave In Sent
sLE :   Leave Empty Sent        sLA : Leave All Sent

Port    rJE    rJIn   rEmp   rLIn   rLE    rLA    sJE    sJIn   sEmp   sLIn   sLE    sLA
-----
gi1     0      0      0      0      0      0      0      0      0      0      0      0
gi2     0      0      0      0      0      0      0      0      0      0      0      0
gi3     0      0      0      0      0      0      0      0      0      0      0      0
gi4     0      0      0      0      0      0      0      0      0      0      0      0
gi5     0      0      0      0      0      0      0      0      0      0      0      0
gi6     0      0      0      0      0      0      0      0      0      0      0      0
gi7     0      0      0      0      0      0      0      0      0      0      0      0
gi8     0      0      0      0      0      0      0      0      0      0      0      0

```

**34.8 show gvrp error-statistics**

Use the **show gvrp error-statistics** EXEC mode command to display GVRP error statistics for all interfaces or for a specific interface.

**Syntax**

**show gvrp error-statistics** *[interface-id]*

**Parameters**

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Default Configuration**

All GVRP error statistics are displayed.

**Command Mode**

EXEC mode

**Example**

The following example displays GVRP error statistics.

---

```

console# show gvrp error-statistics
GVRP Error Statistics:
-----
Legend:
  INVPROT  : Invalid Protocol Id
  INVATYP  : Invalid Attribute Type  INVALEN  : Invalid Attribute Length
  INVAVAL  : Invalid Attribute Value INVEVENT: Invalid Event

  Port    INVPROT  INVATYP  INVAVAL  INVALEN  INVEVENT
-----
gi1      0         0         0         0         0
gi2      0         0         0         0         0
gi3      0         0         0         0         0
gi4      0         0         0         0         0
gi5      0         0         0         0         0
gi6      0         0         0         0         0
gi7      0         0         0         0         0
gi8      0         0         0         0         0

```

## 35 IP Addressing Commands

### 35.1 ip address

Use the **ip address** Interface Configuration (Ethernet, VLAN, Port-channel) mode command to define an IP address for an interface. Use the **no** form of this command to remove an IP address definition.

#### Syntax

If the product is a switch router.

```
ip address ip-address {mask | prefix-length}
```

```
no ip address [ip-address]
```

If the product is a switch only.

```
ip address ip-address {mask | /prefix-length} [default-gateway ip-address]
```

```
no ip address [ip-address]
```

If the product is switch only and supports a single IP address:

```
ip address ip-address {mask | /prefix-length} [default-gateway ip-address]
```

```
no ip address
```

#### Parameters

- **ip-address**—Specifies the IP address.
- **mask**—Specifies the network mask of the IP address.
- **prefix-length**—Specifies the number of bits that comprise the IP address prefix. The prefix length must be preceded by a forward slash (/). (Range: 8–30)
- **default-gateway ip-address**—Specifies the default gateway IP address.

#### Default Configuration

No IP address is defined for interfaces.

#### Command Mode

Interface Configuration (Ethernet, VLAN, Port-channel) mode. It cannot be configured for a range of interfaces (range context).

### User Guidelines

Defining a static IP address on an interface implicitly removes the DHCP client configuration on the interface.

If the product supports multiple IP addresses:

The product supports up to x IP addresses. The IP addresses should be from different IP subnets. When adding an IP address from a subnet that already exists in the list, the new IP address replaces the existing IP address from that subnet.

If the product is switch only and supports a single IP address.

If the Paddlers configured in global context then it would be bound to the currently defined management interface. If the management interface is Default VLAN and the VID of the default VLAN is changed then when new setting is applied, the IP address will be automatically redefined on the new Default VLAN.

If the IP address is configured in Interface context then the IP address is bound to the interface in context.

If a static IP address is already defined, the user must do **no IP address** in the relevant interface context before changing the IP address.

If a dynamic IP address is already defined, the user must do **no ip address** in the relevant interface context before configuring another dynamic IP address.

The Interface context could be a port, LAG or VLAN, depending on support that is defined for the product.

### Example

The following example configures VLAN 1 with IP address 131.108.1.27 and subnet mask 255.255.255.0.

```
Console(config)# interface vlan 1
Console(config-if)# ip address 131.108.1.27 255.255.255.0
```

## 35.2 ip address dhcp

Use the **ip address dhcp** Interface Configuration (Ethernet, VLAN, Port-channel) mode command to acquire an IP address for an Ethernet interface from the Dynamic Host Configuration Protocol (DHCP) server. Use the **no** form of this command to release an acquired IP address.

### Syntax

**ip address dhcp**

**no ip address dhcp**

### Parameters

No parameters

### Command Mode

Interface Configuration (Ethernet, VLAN, Port-channel) mode. It cannot be configured for a range of interfaces (range context).

### User Guidelines

The **ip address dhcp** command allows any interface to dynamically learn its IP address by using the DHCP protocol.

DHCP client configuration on an interface implicitly removes the static IP address configuration on the interface.

If the device is configured to obtain its IP address from a DHCP server, it sends a DHCPDISCOVER message to provide information about itself to the DHCP server on the network.

If the **ip address dhcp** command is used with or without the optional keyword, the DHCP option 12 field (host name option) is included in the DISCOVER message. By default, the host name specified in the option 12 field is the globally configured device host name.

The **no ip address dhcp** command releases any IP address that was acquired, and sends a DHCPRELEASE message.

### Example

The following example acquires an IP address for `gi16` from DHCP.

---

```
Console(config)# interface gi16
Console(config-if)# ip address dhcp
```

---



---

### 35.3 renew dhcp

Use the **renew dhcp** Privileged EXEC mode command to renew an IP address that was acquired from a DHCP server for a specific interface.

#### Syntax

```
renew dhcp {interface-id} [force-autoconfig]
```

#### Parameters

- **interface-id**—This parameter is only available when the device is in Layer 3 (routing mode). Specifies an interface ID. The interface ID can be one of the following types: Ethernet port, Port-channel or VLAN.
- **force-autoconfig** - If the DHCP server holds a DHCP option 67 record for the assigned IP address, the record overwrites the existing device configuration.

#### Command Mode

Privileged EXEC mode

#### User Guidelines

Note that this command does not enable DHCP on an interface. If DHCP is not enabled on the requested interface, the command returns an error message.

If DHCP is enabled on the interface and an IP address was already acquired, the command tries to renew that IP address.

If DHCP is enabled on the interface and an IP address has not yet been acquired, the command initiates a DHCP request.

#### Example

The following example renews an IP address that was acquired from a DHCP server for VLAN 19. This assumes that the device is in Layer 3.

---

```
Console# renew dhcp vlan 19
```

---

## 35.4 ip default-gateway

The **ip default-gateway** Global Configuration mode command defines a default gateway (device). Use the **no** form of this command to restore the default configuration.

### Syntax

**ip default-gateway** *ip-address*

**no ip default-gateway**

### Parameters

**ip-address**—Specifies the default gateway IP address.

### Command Mode

Global Configuration mode

### Default Configuration

No default gateway is defined.

### Example

The following example defines default gateway 192.168.1.1.

---

```
Console(config)# ip default-gateway 192.168.1.1
```

---

## 35.5 show ip interface

Use the **show ip interface** EXEC mode command to display the usability status of configured IP interfaces.

### Syntax

**show ip interface** [*interface-id*]

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port, Port-channel or VLAN.

**Command Mode**

EXEC mode

**Example**

The following example displays the configured IP interfaces and their types.

The information on the default gateway is not shown when the device is in router mode

---

```
console# show ip interface
```

Gateway IP Address	Activity status	Type
1.1.1.254	Inactive	Static

IP Address	I/F	Type	Status
1.1.1.1/8	vlan 1	Static	Valid
2.2.2.2/24	gil	Static	Valid

**35.6 arp**

Use the **arp** Global Configuration mode command to add a permanent entry to the Address Resolution Protocol (ARP) cache. Use the **no** form of this command to remove an entry from the ARP cache.

**Syntax**

```
arp ip-address mac-address [interface-id]
```

```
no arp ip-address
```

**Parameters**

- **ip-address**—IP address or IP alias to map to the specified MAC address.
- **mac-address**—MAC address to map to the specified IP address or IP alias.
- **interface-id**—interface ID. Can be Ethernet port, Port-channel or VLAN.

**Command Mode**

Global Configuration mode

**Default Configuration**

No permanent entry is defined.

**User Guidelines**

The software uses ARP cache entries to translate 32-bit IP addresses into 48-bit hardware (MAC) addresses. Because most hosts support dynamic address resolution, static ARP cache entries generally do not need to be specified.

**Example**

The following example adds IP address 198.133.219.232 and MAC address 00:00:0c:40:0f:bc to the ARP table.

---

```
Console(config)# arp 198.133.219.232 00:00:0c:40:0f:bc gi6
```

---

## 35.7 arp timeout (Global)

Use the **arp timeout** Global Configuration mode command to set the time interval during which an entry remains in the ARP cache. Use the **no** form of this command to restore the default configuration.

**Syntax**

**arp timeout** *seconds*

**no arp timeout**

**Parameters**

**seconds**—Specifies the time interval (in seconds) during which an entry remains in the ARP cache.  
(Range: 1–40000000)

**Default Configuration**

The default ARP timeout is 60000 seconds in Router mode, and 300 seconds in Switch mode.

**Command Mode**

Global Configuration mode

**Example**

The following example configures the ARP timeout to 12000 seconds.

---

```
Console(config)# arp timeout 12000
```

---

**35.8 ip arp proxy disable**

Use the **ip arp proxy disable** Global Configuration mode command to globally disable proxy Address Resolution Protocol (ARP). Use the **no** form of this command reenables proxy ARP.

This command can be used in Router mode only.

**Syntax**

**ip arp proxy disable**

**no ip arp proxy disable**

**Parameters**

N/A

**Default**

Enabled by default.

**Command Mode**

Global Configuration mode

**User Guidelines**

The **ip arp proxy disable** command overrides any proxy ARP interface configuration. To use this command, you must put the switch into routing mode using the **set system mode router** command.

### Example

The following example globally disables ARP proxy when the switch is in router mode.

---

```
Console (config)# ip arp proxy disable
```

---

## 35.9 ip proxy-arp

Use the **ip proxy-arp** Interface Configuration mode command to enable an ARP proxy on specific interfaces. Use the **no** form of this command disable it.

### Syntax

**ip proxy-arp**

**no ip proxy-arp**

### Default Configuration

ARP Proxy is disabled.

### Command Mode

Interface Configuration (Ethernet, VLAN, Port-channel) mode. It cannot be configured for a range of interfaces (range context).

### User Guidelines

This configuration can be applied only if at least one IP address is defined on a specific interface. To use this command, you must put the switch into routing mode using the **set system mode router** command.

### Example

The following example enables ARP proxy when the switch is in router mode.

---

```
Console(config-if)# ip proxy-arp
```

---

## 35.10 clear arp-cache

Use the **clear arp-cache** Privileged EXEC mode command to delete all dynamic entries from the ARP cache.

**Syntax****clear arp-cache****Command Mode**

Privileged EXEC mode

**Example**

The following example deletes all dynamic entries from the ARP cache.

---

```
Console# clear arp-cache
```

---

**35.11 show arp**

Use the **show arp** Privileged EXEC mode command to display entries in the ARP table.

**Syntax****show arp** [*ip-address ip-address*] [*mac-address mac-address*] [*interface-id*]**Parameters**

- **ip-address** *ip-address*—Specifies the IP address.
- **mac-address** *mac-address*—Specifies the MAC address.
- **interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

**Command Mode**

Privileged EXEC mode

**User Guidelines**

Since the associated interface of a MAC address can be aged out from the FDB table, the Interface field can be empty.

If an ARP entry is associated with an IP interface that is defined on a port or port-channel, the VLAN field is empty.

### Example

The following example displays entries in the ARP table.

---

```
Console# show arp

ARP timeout: 80000 Seconds

VLAN      Interface  IP Address  HW Address  Status
-----  -
VLAN 1    gi1        10.7.1.102  00:10:B5:04:DB:4B  Dynamic
VLAN 1    gi2        10.7.1.135  00:50:22:00:2A:A4  Static
```

---

## 35.12 show arp configuration

Use the **show arp configuration** privileged EXEC command to display the global and interface configuration of the ARP protocol.

### Syntax

**show arp configuration**

### Parameters

N/A

### Command Mode

Privileged EXEC mode

### Example

---

```
Console# show arp configuration

Global configuration:

ARP Proxy: enabled

ARP timeout:      80000 Seconds

Interface configuration:
```



```
g2:
  ARP Proxy: disabled
  ARP timeout:60000 Seconds

VLAN 1:
  ARP Proxy: enabled
  ARP timeout:70000 Seconds

VLAN 2:
  ARP Proxy: enabled
  ARP timeout:80000 Second (Global)
```

---

## 35.13 interface ip

Use the **interface ip** Global Configuration mode command to enter the IP Interface Configuration mode.

### Syntax

**interface ip** *address*

### Parameters

**ip-address**—Specifies one of the IP addresses of the device.

### Command Mode

Global Configuration mode

### User Guidelines

To use this command, you must put the switch into routing mode using the **set system mode router** command.

### Example

The following example enters the IP interface configuration mode.

---

```
Console (config)# interface ip 192.168.1.1
```

```
Console (config-ip)#
```

---

## 35.14 ip helper-address

Use the **ip helper-address** Global Configuration mode command to enable the forwarding of User Datagram Protocol (UDP) broadcast packets received on an interface to a specific (helper) address. Use the **no** form of this command to disable the forwarding of broadcast packets to a specific (helper) address.

### Syntax

```
ip helper-address {ip-interface / all} address [udp-port-list]
```

```
no ip helper-address {ip-interface / all} address
```

### Parameters

- **ip-interface**—Specifies the IP interface.
- **all**—Specifies all IP interfaces.
- **address**—Specifies the destination broadcast or host address to which to forward UDP broadcast packets. A value of 0.0.0.0 specifies that UDP broadcast packets are not forwarded to any host.
- **udp-port-list**—Specifies the destination UDP port number to which to forward broadcast packets. (Range: 1–65535)

### Default Configuration

Forwarding of User Datagram Protocol (UDP) broadcast packets received on an interface to a specific (helper) address is disabled.

If **udp-port-list** is not specified, packets for the default services are forwarded to the helper address.

### Command Mode

Global Configuration mode

### User Guidelines

To use this command, you must put the switch into routing mode using the **set**

**system mode router command.**

The **ip helper-address** command forwards specific UDP broadcast packets from one interface to another.

Many helper addresses may be defined. However, the total number of address-port pairs is limited to 128 for the device.

The setting of a helper address for a specific interface has precedence over the setting of a helper address for all the interfaces.

Forwarding of BOOTP/DHCP (ports 67, 68) cannot be enabled with this command. Use the DHCP relay commands to relay BOOTP/DHCP packets.

The **ip helper-address** command specifies a UDP port number to which UDP broadcast packets with that destination port number are forwarded. By default, if no UDP port number is specified, the device forwards UDP broadcast packets for the following six services:

- IEN-116 Name Service (port 42)
- DNS (port 53)
- NetBIOS Name Server (port 137)
- NetBIOS Datagram Server (port 138)
- TACACS Server (port 49)
- Time Service (port 37)

**Example**

The following example enables the forwarding of User Datagram Protocol (UDP) broadcasts received on all interfaces to specific UDP ports of a destination IP address.

---

```
Console (config)# ip helper-address all 172.16.9.9 49 53
```

---

**35.15 show ip helper-address**

Use the **show ip helper-address** Privileged EXEC mode command to display the IP helper addresses configuration on the system.

**Syntax**

**show ip helper-address**

**Parameters**

N/A

**Command Mode**

Privileged EXEC mode

**User Guidelines**

To use this command, you must put the switch into routing mode using the **set system mode router** command.

**Example**

The following example displays the IP helper addresses configuration on the system.

---

```
Console# show ip helper-address
```

Interface	Helper Address	Udp ports
-----	-----	-----
192.168.1.1	172.16.8.8	37, 42, 49, 53, 137, 138
192.168.2.1	172.16.9.9	37, 49

**35.16 ip domain name**

Use the **ip domain name** Global Configuration mode command to define a default domain name used by the software to complete unqualified host names (names without a dotted-decimal domain name). Use the **no** form of this command to remove the default domain name.

**Syntax**

**ip domain name** *name*

**no ip domain name**

**Parameters**

**name**—Specifies the default domain name used to complete unqualified host names. Do not include the initial period that separates an unqualified name from the domain name. (Length: 1–158 characters. Maximum label length: 63 characters)

### Default Configuration

A default domain name is not defined.

### Command Mode

Global Configuration mode

### User Guidelines

Domain names and host names are restricted to the ASCII letters A through Z (case-insensitive), the digits 0 through 9, the underscore and the hyphen. A period (.) is used to separate labels.

The maximum size of a label is 63 characters. The maximum name size is 158 bytes.

### Example

The following example defines the default domain name as 'www.website.com'.

---

```
Console(config)# ip domain name www.website.com
```

---

## 35.17 ip name-server

Use the **ip name-server** Global Configuration mode command to define the available name servers. Use the **no** form of this command to remove a name server.

### Syntax

```
ip name-server {server1-ipv4-address | server1-ipv6-address} [server-address2 ...  
server-address8]
```

```
no ip name-server [server-address ... server-address8]
```

### Parameters

**server-address**—IP addresses of the name server. Up to 8 servers can be defined in one command or by using multiple commands. The IP address can be IPv4 address or IPv6 address. When the IPv6 address is a Link Local address (IPv6Z address), the outgoing interface name must be specified. Refer to the User Guidelines for the interface name syntax.

### Default Configuration

No name server IP addresses are defined.

### Command Mode

Global Configuration mode

### User Guidelines

The preference of the servers is determined by the order in which they were entered.

Up to 8 servers can be defined using one command or using multiple commands.

The format of an **IPv6Z address** is: <ipv6-link-local-address>%<interface-name>

**interface-name** = vlan<integer> | ch<integer> | isatap<integer> | <physical-port-name> | 0

**integer** = <decimal-number> | <integer><decimal-number>

**decimal-number** = 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

physical-port-name= Designated port number, for example gi16.

If the egress interface is not specified, the default interface is selected. Specifying interface zone=0 is equal to not defining an egress interface.

### Example

The following example defines the available name server.

---

```
Console(config)# ip name-server 176.16.1.18
```

---

## 35.18 ip host

Use the **ip host** Global Configuration mode command to define the static host name-to-address mapping in the host cache. Use the **no** form of this command to remove the static host name-to-address mapping.

### Syntax

**ip host** *name address [address2 address3 address4]*

**no ip host** *name*

### Parameters

- **name**—Specifies the host name. (Length: 1–158 characters. Maximum label length: 63 characters)
- **address**—Specifies the associated IP address. Up to 4 addresses can be defined.

### Default Configuration

No host is defined.

### Command Mode

Global Configuration mode

### User Guidelines

Host names are restricted to the ASCII letters A through Z (case-insensitive), the digits 0 through 9, the underscore and the hyphen. A period (.) is used to separate labels.

### Example

The following example defines a static host name-to-address mapping in the host cache.

---

```
Console(config)# ip host accounting.website.com 176.10.23.1
```

---

## 35.19 clear host

Use the **clear host** Privileged EXEC mode command to delete entries from the host name-to-address cache.

### Syntax

```
clear host {name / *}
```

### Parameters

- **name**—Specifies the host entry to remove. (Length: 1–158 characters. Maximum label length: 63 characters)
- **\*** —Removes all entries.

### Command Mode

Privileged EXEC mode

### Example

The following example deletes all entries from the host name-to-address cache.

---

```
Console# clear host *
```

---

## 35.20 clear host dhcp

Use the **clear host dhcp** Privileged EXEC mode command to delete entries from the host name-to-address mapping received from Dynamic Host Configuration Protocol (DHCP).

### Syntax

```
clear host dhcp {name / *}
```

### Parameters

- **name** —Specifies the host entry to remove. (Length: 1–158 characters. Maximum label length: 63 characters)
- **\***—Removes all entries.

### Command Mode

Privileged EXEC mode

### User Guidelines

This command deletes the host name-to-address mapping temporarily until the next refresh of the IP addresses.

### Example

The following example deletes all entries from the host name-to-address mapping received from DHCP.

---

```
Console# clear host dhcp *
```



---

## 35.21 show hosts

Use the **show hosts** EXEC mode command to display the default domain name, the list of name server hosts, the static and the cached list of host names and addresses.

### Syntax

**show hosts** *[name]*

### Parameters

**name**—Specifies the host name. (Length: 1–158 characters. Maximum label length: 63 characters)

### Command Mode

EXEC mode

### Example

The following example displays host information.

---

```
Console> show hosts

System name: Device

Default domain is gm.com, sales.gm.com, usa.sales.gm.com(DHCP)

Name/address lookup is enabled

Name servers (Preference order): 176.16.1.18 176.16.1.19

Configured host name-to-address mapping:
Host                               Addresses
-----
accounting.gm.com                  176.16.8.8 176.16.8.9 (DHCP)
                                     2002:0:130F::0A0:1504:0BB4
```

---

Host	Total	Elapsed	Type	Addresses
-----	-----	-----	-----	-----
www.stanford.edu	72	3	IP	171.64.14.203

---

## 36 IPv6 Addressing Commands

---

### 36.1 ipv6 enable

Use the **ipv6 enable** Interface Configuration (Ethernet, VLAN, Port-channel) mode command to enable the IPv6 addressing mode on an interface. Use the **no** form of this command to disable the IPv6 addressing mode on an interface.

#### Syntax

**ipv6 enable** [*no-autoconfig*]

**no ipv6 enable**

#### Parameters

**no-autoconfig**—EnableS processing of IPv6 on an interface without stateless address autoconfiguration procedure

#### Default Configuration

IPv6 addressing is disabled.

Unless you are using the no-autoconfig parameter, when the interface is enabled stateless address autoconfiguration procedure is enabled.

#### Command Mode

Interface Configuration (Ethernet, VLAN, Port-channel) mode. It cannot be configured for a range of interfaces (range context).

#### User Guidelines

The **ipv6 enable** command automatically configures an IPv6 link-local unicast address on the interface, while also enabling the interface for IPv6 processing. The **no ipv6 enable** command removes the entire IPv6 interface configuration.

To enable stateless address autoconfiguration on an enabled IPv6 interface, use the IPv6 address autoconfig command.

#### Example

The following example enables VLAN 1 for the IPv6 addressing mode.

---

```
Console(config)# interface vlan 1
```

```
Console(config-if)# ipv6 enable
```

---

## 36.2 ipv6 address autoconfig

Use the **ipv6 address autoconfig** Interface Configuration mode command to enable automatic configuration of IPv6 addresses, using stateless autoconfiguration on an interface. Addresses are configured depending on the prefixes received in Router Advertisement messages. Use the **no** form of this command to disable address autoconfiguration on the interface.

### Syntax

```
ipv6 address autoconfig
```

```
no ipv6 address autoconfig
```

### Parameters

N/A

### Default Configuration

Address autoconfiguration is enabled on the interface, no addresses are assigned by default.

### Command Mode

Interface Configuration (Ethernet, VLAN, Port-channel) mode.

### User Guidelines

When **address autoconfig** is enabled, router solicitation ND procedure is initiated to discover a router and assign IP addresses to the interface, based on the advertised on-link prefixes.

When disabling address autoconfig, automatically generated addresses that are assigned to the interface are removed.

The default state of the address autoconfig is enabled. Use the **enable ipv6 no-autoconfig** command to enable an IPv6 interface without address autoconfig.

### Example

---

```
console(config)# interface vlan 1  
console(config-if)# ipv6 address autoconfig
```

### 36.3 ipv6 icmp error-interval

Use the **ipv6 icmp error-interval** Global Configuration mode command to configure the rate limit interval and bucket size parameters for IPv6 Internet Control Message Protocol (ICMP) error messages. Use the **no** form of this command to return the interval to its default setting.

#### Syntax

**ipv6 icmp error-interval** *milliseconds* [*bucketsize*]

**no ipv6 icmp error-interval**

#### Parameters

- **milliseconds**—The time interval between tokens being placed in the bucket. Each token represents a single ICMP error message. The acceptable range is from 0–2147483647 with a default of 100 milliseconds. Setting milliseconds to 0 disables rate limiting. (Range: 0– 2147483647)
- **bucketsize**—(Optional) The maximum number of tokens stored in the bucket. The acceptable range is from 1–200 with a default of 10 tokens.

#### Default Configuration

The default interval is 100ms and the default bucketsize is 10 i.e. 100 ICMP error messages per second

#### Command Mode

Global Configuration mode

#### User Guidelines

To set the average ICMP error rate limit, calculate the interval with the following formula:

Average Packets Per Second = (1/ interval) \* bucket size

#### Example

```
console(config)# ipv6 icmp error-interval 123 45
```

---

## 36.4 show ipv6 icmp error-interval

Use the **show ipv6 error-interval** command in the EXEC mode to display the IPv6 ICMP error interval.

### Syntax

**show ipv6 icmp error-interval**

### Command Mode

EXEC mode

### Example

---

```
Console> show ipv6 icmp error-interval
```

```
Rate limit interval: 100 ms
```

```
Bucket size: 10 tokens
```

---

## 36.5 ipv6 address

Use the **ipv6 address** Interface Configuration mode command to configure an IPv6 address for an interface. Use the **no** form of this command To remove the address from the interface.

### Syntax

**ipv6 address** *ipv6-address/prefix-length*

**no ipv6 address** [*ipv6-address/prefix-length*]

### Parameters

- **ipv6-address**—Specifies the IPv6 network assigned to the interface. This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.
- **prefix-length**—Specifies the length of the IPv6 prefix. A decimal value that indicates how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark (/) must precede the decimal.
- **eui-64**—(Optional) Builds an interface ID in the low order 64 bits of the IPv6 address based on the interface MAC address.

- **anycast**—(Optional) Indicates that this address is an anycast address.
- **prefix-length**—3–128 (64 when the **eui-64** parameter is used).

### Default Configuration

No IP address is defined for the interface.

### Command Mode

Interface configuration (Ethernet, VLAN, Port-channel) mode. It cannot be configured for a range of interfaces (range context).

### User Guidelines

If the value specified for the /prefix-length argument is greater than 64 bits, the prefix bits have precedence over the interface ID.

Using the no IPv6 address command without arguments removes all manually configured IPv6 addresses from an interface, including link local manually configured addresses.

### Example

```
console(config)# interface vlan 1
console(config-if)# ipv6 address 3000::123/64 eui-64 anycast
```

## 36.6 ipv6 address link-local

Use the **ipv6 address link-local** command to configure an IPv6 link-local address for an interface. Use the **no** form of this command to return to the default link local address on the interface.

### Syntax

```
ipv6 address ipv6-address/prefix-length link-local
```

```
no ipv6 address [ipv6-address/prefix-length link-local]
```

### Parameters

- **ipv6-address**—Specifies the IPv6 network assigned to the interface. This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimals using 16-bit values between colons.

- **prefix-length**—Specifies the length of the IPv6 prefix. A decimal value indicates how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark (/) must precede the decimal. Only 64-bit length is supported, according to IPv6 over Ethernet's well-known practice

### Default Configuration

IPv6 is enabled on the interface, link local address of the interface is FE80::EUI64 (interface MAC address).

### Command Mode

Interface Configuration (Ethernet, VLAN, Port-channel) mode. It cannot be configured for a range of interfaces (range context).

### User Guidelines

Using the **no ipv6 link-local address** command removes the manually configured link local IPv6 address from an interface. Multiple IPv6 addresses can be configured per interface, but only one link-local address. When the **no ipv6 link-local address** command is used, the interface is reconfigured with the standard link local address (the same IPv6 link-local address that is set automatically when the **enable ipv6** command is used). The system automatically generates a link-local address for an interface when IPv6 processing is enabled on the interface. To manually specify a link-local address to be used by an interface, use the **ipv6 link-local address** command. The system supports only 64 bits prefix length for link-local addresses.

### Example

```
console(config)# interface vlan 1
console(config-if)# ipv6 address fe80::123/64 link-local
```

---

## 36.7 ipv6 unreachable

Use the **ipv6 unreachable** Interface Configuration mode command to enable the generation of Internet Control Message Protocol for IPv6 (ICMPv6) unreachable messages for any packets arriving on a specified interface. Use the **no** form of this command To prevent the generation of unreachable messages.

### Syntax

**ipv6 unreachable**



## **no ipv6 unreachable**

### **Parameters**

N/A

### **Default Configuration**

ICMP unreachable messages are sent by default.

### **Command Mode**

Interface Configuration (Ethernet, VLAN, Port-channel) mode.

### **User Guidelines**

When ICMP unreachable messages are enabled, when receiving a packet addressed to one of the interface's IP address with TCP/UDP port not assigned, the device sends ICMP unreachable messages.

### **Example**

```
console(config)# interface gil
console(config-if)# ipv6 unreachable
```

---

## **36.8 ipv6 default-gateway**

Use the **ipv6 default-gateway** Global Configuration mode command to define an IPv6 default gateway. Use the **no** form of this command To remove the default gateway.

### **Syntax**

**ipv6 default-gateway** *ipv6-address*

**no ipv6 default-gateway**

### **Parameters**

**ipv6-address**—Specifies the IPv6 address of the next hop that can be used to reach that network. When the IPv6 address is a Link Local address (IPv6Z address), the outgoing interface name must be specified. Refer to the User Guidelines for the interface ID syntax.

### Default Configuration

No default gateway is defined.

### Command Mode

Global Configuration mode

### User Guidelines

The format of an IPv6Z address is: {ipv6-link-local-address}>%<*interface-id*}

Configuring a new default GW without deleting the previous configured information overwrites the previous configuration. A configured default GW has a higher precedence over automatically advertised (via router advertisement message). Precedence takes effect once the configured default GW is reachable. Reachability state is not verified automatically by the neighbor discovery protocol. Router reachability can be confirmed by either receiving Router Advertisement message containing router's MAC address or manually configured by user using the IPv6 neighbor CLI command. Another option to force reachability confirmation is to ping the router link-local address (this will initiate the neighbor discovery process).

If the egress interface is not specified, the default interface is selected. Specifying interface zone=0 is equal to not defining an egress interface.

### Example

```
console(config)# ipv6 default-gateway fe80::abcd
```

---

## 36.9 show ipv6 interface

Use the **show ipv6 interface** EXEC command mode to display the usability status of interfaces configured for IPv6.

### Syntax

```
show ipv6 interface [interface-id]
```

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port, Port-channel or VLAN.

## Default Configuration

Displays all IPv6 interfaces.

## Command Mode

EXEC mode

## User Guidelines

Use the **show ipv6 neighbors** command in the privileged EXEC mode to display IPv6 neighbor discovery cache information.

## Example

```
Console# show ipv6 interface
```

Interface	IP addresses	Type
VLAN 1	4004::55/64 [ANY]	manual
VLAN 1	fe80::200:b0ff:fe00:0	linklayer
VLAN 1	ff02::1	linklayer
VLAN 1	ff02::77	manual
VLAN 1	ff02::1:ff00:0	manual
VLAN 1	ff02::1:ff00:1	manual
VLAN 1	ff02::1:ff00:55	manual

Default Gateway IP address	Type	Interface	State
fe80::77	Static	VLAN 1	unreachable
fe80::200:cff:fe4a:dfa8	Dynamic	VLAN 1	stale

```
Console# show ipv6 interface Vlan 15
```

```
IPv6 is disabled
```

```
Console# show ipv6 interface Vlan 1
```

```
Number of ND DAD attempts: 1
```

```

MTU size: 1500
Stateless Address Autoconfiguration state: enabled
ICMP unreachable message state: enabled
MLD version: 2

```

IP addresses	Type	DAD State
4004::55/64 [ANY]	manual	Active
fe80::200:b0ff:fe00:0	linklayer	Active
ff02::1	linklayer	-----
ff02::77	manual	-----
ff02::1:ff00:0	manual	-----
ff02::1:ff00:1	manual	-----
ff02::1:ff00:55	manual	-----

## 36.10 show IPv6 route

Use the **show ipv6 route** command to display the current state of the IPv6 routing table.

### Syntax

```
show ipv6 route
```

### Command Mode

EXEC mode

### Example

```

Console> show ipv6 route

Codes: L - Local, S - Static, I - ICMP, ND - Router Advertisement

The number in the brackets is the metric.

S  ::/0 via fe80::77 [0] VLAN 1 Lifetime Infinite

```

```
ND ::/0 via fe80::200:cff:fe4a:dfa8 [0] VLAN 1 Lifetime 1784 sec
L 2001::/64 is directly connected, g2 Lifetime Infinite
L 2002:1:1:1::/64 is directly connected, VLAN 1 Lifetime 2147467 sec
L 3001::/64 is directly connected, VLAN 1 Lifetime Infinite
L 4004::/64 is directly connected, VLAN 1 Lifetime Infinite
L 6001::/64 is directly connected, g2 Lifetime Infinite
```

---

### 36.11 ipv6 nd dad attempts

Use the **ipv6 nd dad attempts** Interface Configuration (Ethernet, VLAN, Port-channel) mode command to configure the number of consecutive neighbor solicitation messages that are sent on an interface while Duplicate Address Detection (DAD) is performed on the unicast IPv6 addresses of the interface. Use the **no** form of this command to restore the number of messages to the default value.

#### Syntax

**ipv6 nd dad attempts** *attempts*

#### Parameters

**attempts**—Specifies the number of neighbor solicitation messages. A value of 0 disables DAD processing on the specified interface. A value of 1 configures a single transmission without follow-up transmissions. (Range: 0–600)

#### Default Configuration

Duplicate Address Detection on unicast IPv6 addresses with the sending of one neighbor solicitation message is enabled.

#### Command Mode

Interface Configuration (Ethernet, VLAN, Port-channel) mode. It cannot be configured for a range of interfaces (range context).

#### User Guidelines

Duplicate Address Detection (DAD) verifies the uniqueness of new unicast IPv6 addresses before the addresses are assigned to interfaces (the new addresses remain in a tentative state while DAD is performed). DAD uses neighbor solicitation messages to verify the uniqueness of unicast IPv6 addresses.

An interface returning to the administrative Up state restarts DAD for all of the unicast IPv6 addresses on the interface. While DAD is performed on the Link Local address of an interface, the state of the other IPv6 addresses is still set to TENTATIVE. When DAD is completed on the Link Local address, DAD is performed on the remaining IPv6 addresses.

When DAD identifies a duplicate address, the address state is set to DUPLICATE and the address is not used. If the duplicate address is the Link Local address of the interface, the processing of IPv6 packets is disabled on the interface and an error message is displayed.

All configuration commands associated with the duplicate address remain as configured while the address state is set to DUPLICATE.

If the Link Local address for an interface changes, DAD is performed on the new Link Local address and all of the other IPv6 address associated with the interface are regenerated (DAD is performed only on the new Link Local address).

Configuring a value of 0 with the **ipv6 nd dad attempts** Interface Configuration mode command disables duplicate address detection processing on the specified interface. A value of 1 configures a single transmission without follow-up transmissions. The default is 1 message.

Until the DAD process is completed, an IPv6 address is in the tentative state and cannot be used for data transfer. It is recommended to limit the configured value.

### Example

The following example configures the number of consecutive neighbor solicitation messages sent during DAD processing to 2 on `gi9`.

```
Console (config)# interface gi9
Console (config-if)# ipv6 nd dad attempts 2
```

---

## 36.12 ipv6 host

Use the **ipv6 host** Global Configuration mode command to define a static host name-to-address mapping in the host name cache. Use the **no** form of this command to remove the host name-to-address mapping.

### Syntax

```
ipv6 host name ipv6-address1 [ipv6-address2...ipv6-address4]
```

```
no ipv6 host name
```

### Parameters

**host name** - Name of the host. (Range: 1–158 characters)

- **ipv6-address1**—Associated IPv6 address. This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. When the IPv6 address is a Link Local address (IPv6Z address), the outgoing interface name must be specified. Refer to the user guidelines for the interface name syntax.
- **ipv6-address2-4**—(Optional) Additional IPv6 addresses that may be associated with the host's name

### Default Configuration

No host is defined.

### Command Mode

Global Configuration mode

### User Guidelines

The format of an IPv6Z address is: <ipv6-link-local-address>%<interface-id

### Example

```
console(config)# ipv6 host server 3000::a31b
```

---

## 36.13 ipv6 neighbor

Use the **ipv6 neighbor** command to configure a static entry in the IPv6 neighbor discovery cache. Use the **no** form of this command to remove a static IPv6 entry from the IPv6 neighbor discovery cache.

### Syntax

```
ipv6 neighbor ipv6_addr interface-id hw_addr
```

```
no ipv6 neighbor ipv6_addr interface-id
```

### Parameters

- **ipv6\_addr**—Specifies the Pv6 address to map to the specified MAC address.

- **interface-id**—Specifies the interface that is associated with the IPv6 address
- **hw\_addr**—Specifies the MAC address to map to the specified IPv6 address.

### Command Mode

Global Configuration mode

### User Guidelines

The **IPv6 neighbor** command is similar to the **ARP** (global) command.

If an entry for the specified IPv6 address already exists in the neighbor discovery cache, learned through the IPv6 neighbor discovery process, the entry is automatically converted to a static entry.

A new static neighbor entry with a global address can be configured only if a manually configured subnet already exists in the device.

Use the **show IPv6 neighbors** command to view static entries in the IPv6 neighbor discovery cache.

### Example

```
console(config)# ipv6 neighbor 3000::a31b vlan 1 001b.3f9c.84ea
```

## 36.14 ipv6 set mtu

Use the **ipv6 mtu** Interface Configuration mode command to set the maximum transmission unit (MTU) size of IPv6 packets sent on an interface. Use the default parameter to restore the default MTU size.

### Syntax

```
ipv6 set mtu {interface-id} {bytes | default}
```

### Parameters

- **interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.
- **bytes**—Specifies the MTU in bytes.
- **default**—Sets the default MTU size 1500 bytes. Minimum is 1280 bytes



### Default Configuration

1500 bytes

### Command Mode

Privileged EXEC mode

### User Guidelines

This command is intended for debugging and testing purposes and should be used only by technical support personnel.

### Example

```
console# ipv6 set mtu gil default
```

---

## 36.15 show ipv6 neighbors

Use the **show ipv6 neighbors** Privileged EXEC mode command to display IPv6 neighbor discovery cache information.

### Syntax

```
show ipv6 neighbors {static / dynamic} [ipv6-address ipv6-address] [mac-address mac-address] [interface-id]
```

### Parameters

- **static**—Shows static neighbor discovery cache entries.
- **dynamic**—Shows dynamic neighbor discovery cache entries.
- **ipv6-address**—Shows the neighbor discovery cache information entry of a specific IPv6 address.
- **mac-address**—Shows the neighbor discovery cache information entry of a specific MAC address.
- **interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port, Port-channel or VLAN.

### Command Mode

Privileged EXEC mode

### User Guidelines

Since the associated interface of a MAC address can be aged out from the FDB table, the Interface field can be empty.

When an ARP entry is associated with an IP interface that is defined on a port or port-channel, the VLAN field is empty.

The possible neighbor cash states are:

- **INCOMP (Incomplete)**—Address resolution is being performed on the entry. Specifically, a Neighbor Solicitation has been sent to the solicited-node multicast address of the target, but the corresponding Neighbor Advertisement has not yet been received.
- **REACH (Reachable)**—Positive confirmation was received within the last ReachableTime milliseconds that the forward path to the neighbor was functioning properly. While REACHABLE, no special action takes place as packets are sent.
- **STALE**—More than ReachableTime milliseconds have elapsed since the last positive confirmation was received that the forward path was functioning properly. While stale, no action takes place until a packet is sent.
- **DELAY**—More than ReachableTime milliseconds have elapsed since the last positive confirmation was received that the forward path was functioning properly, and a packet was sent within the last DELAY\_FIRST\_PROBE\_TIME seconds. If no reachability confirmation is received within DELAY\_FIRST\_PROBE\_TIME seconds of entering the DELAY state, send a Neighbor Solicitation and change the state to PROBE.
- **PROBE**—A reachability confirmation is actively sought by retransmitting Neighbor Solicitations every RetransTimer milliseconds until a reachability confirmation is received.

### Example

```
Console# show ipv6 neighbors dynamic
```

Interface	IPv6 address	HW address	State	Router
VLAN 1	fe80::200:cff:fe4a:dfa8	00:00:0c:4a:df:a8	stale	yes
VLAN 1	fe80::2d0:b7ff:fe1:264d	00:d0:b7:a1:26:4d	stale	no

### 36.16 clear ipv6 neighbors

Use the **clear ipv6 neighbors** Privileged EXEC mode command to delete all entries in the IPv6 neighbor discovery cache, except for static entries.

#### Syntax

**clear ipv6 neighbors**

#### Parameters

This command has no keywords or arguments.

#### Command Mode

Privileged EXEC mode

#### Example

---

```
console# clear ipv6 neighbors
```

---

## 37 Tunnel Commands

---

### 37.1 interface tunnel

Use the **interface tunnel** Global Configuration mode command to enter the Interface Configuration (Tunnel) mode.

#### Syntax

**interface tunnel** *number*

#### Parameters

**number**—Specifies the tunnel index.

#### Command Mode

Global Configuration mode

#### Example

The following example enters the Interface Configuration (Tunnel) mode.

---

```
Console(config)# interface tunnel 1
Console(config-tunnel)#
```

---

### 37.2 tunnel mode ipv6ip

Use the **tunnel mode ipv6ip** Interface Configuration (Tunnel) mode command to configure an IPv6 transition-mechanism global support mode. Use the **no** form of this command to remove an IPv6 transition mechanism.

#### Syntax

**tunnel mode ipv6ip** *{isatap}*

**no tunnel mode ipv6ip**

#### Parameters

**isatap**—Enables an automatic IPv6 over IPv4 Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) tunnel.

### Default Configuration

The IPv6 transition-mechanism global support mode is disabled.

### Command Mode

Interface Configuration (Tunnel) mode

### User Guidelines

The system can be enabled to ISATAP tunnel. When enabled, an automatic tunnel interface is created on each interface that is assigned an IPv4 address.

Note that on a specific interface (for example, port or VLAN), both native IPV6 and transition-mechanisms can coexist. The host implementation chooses the egress interface according to the scope of the destination IP address (such as ISATAP or native IPv6).

### Example

The following example configures an IPv6 transition mechanism global support mode.

```
Console(config)# interface tunnel 1
Console(config-tunnel)# tunnel mode ipv6ip isatap
```

---

## 37.3 tunnel isatap router

Use the **tunnel isatap router** Interface Configuration (Tunnel) mode command to configure a global string that represents a specific automatic tunnel router domain name. Use the **no** form of this command to remove the string associated with the router domain name and restore the default configuration.

### Syntax

**tunnel isatap router** *router-name*

**no tunnel isatap router**

### Parameters

**router-name**—Specifies the router's domain name.

### Default Configuration

The automatic tunnel router's default domain name is ISATAP.

### Command Mode

Interface Configuration (Tunnel) mode

### User Guidelines

The **ipv6 tunnel routers-dns** command determines the string that the host uses for automatic tunnel router lookup in the IPv4 DNS procedure. By default, the string ISATAP is used for the corresponding automatic tunnel types.

Only one string can represent the automatic tunnel router name per tunnel. Using this command, therefore, overwrites the existing entry.

### Example

The following example configures the global string ISATAP2 as the automatic tunnel router domain name.

---

```
Console(config)# tunnel 1
Console(config-tunnel)# tunnel isatap router ISATAP2
```

---

## 37.4 tunnel source

Use the **tunnel source** Interface Configuration (Tunnel) mode command to set the local (source) IPv4 address of a tunnel interface. The **no** form deletes the tunnel local address.

### Syntax

**tunnel source** *{auto | ipv4-address}*

**no tunnel source**

### Parameters

- **auto**—The system minimum IPv4 address is used as the source address for packets sent on the tunnel interface. If the IPv4 address is changed, then the local address of the tunnel interface is changed too.
- **ip4-address**—Specifies the IPv4 address to use as the source address for packets sent on the tunnel interface. The local address of the tunnel

interface is not changed when the IPv4 address is moved to another interface.

### Default

No source address is defined.

### Command Mode

Interface Configuration (Tunnel) mode

### User Guidelines

The configured source IPv4 address is used for forming the tunnel interface identifier. The interface identifier is set to the 8 least significant bytes of the SIP field of the encapsulated IPv6 tunneled packets.

### Example

```
console(config)# interface tunnel 1
console(config-tunnel)# tunnel source auto
```

---

## 37.5 tunnel isatap query-interval

Use the **tunnel isatap query-interval** Global Configuration mode command to set the time interval between Domain Name System (DNS) queries (before the ISATAP router IP address is known) for the automatic tunnel router domain name. Use the **no** form of this command to restore the default configuration.

### Syntax

**tunnel isatap query-interval** *seconds*

**no tunnel isatap query-interval**

### Parameters

**seconds**—Specifies the time interval in seconds between DNS queries. (Range: 10–3600)

### Default Configuration

The default time interval between DNS queries is 10 seconds.

### Command Mode

Global Configuration mode

### User Guidelines

This command determines the time interval between DNS queries before the ISATAP router IP address is known. If the IP address is known, the robustness level that is set by the **tunnel isatap robustness** Global Configuration mode command determines the refresh rate.

### Example

The following example sets the time interval between DNS queries to 30 seconds.

---

```
Console(config)# tunnel isatap query-interval 30
```

---

## 37.6 tunnel isatap solicitation-interval

Use the **tunnel isatap solicitation-interval** Global Configuration mode command to set the time interval between ISATAP router solicitation messages. Use the **no** form of this command to restore the default configuration.

### Syntax

**tunnel isatap solicitation-interval** *seconds*

**no tunnel isatap solicitation-interval**

### Parameters

**seconds**—Specifies the time interval in seconds between ISATAP router solicitation messages. (Range: 10–3600)

### Default Configuration

The default time interval between ISATAP router solicitation messages is 10 seconds.

### Command Mode

Global Configuration mode



### User Guidelines

This command determines the interval between router solicitation messages when there is no active ISATAP router. If there is an active ISATAP router, the robustness level set by the **tunnel isatap robustness** Global Configuration mode command determines the refresh rate.

### Example

The following example sets the time interval between ISATAP router solicitation messages to 30 seconds.

```
Console(config)# tunnel isatap solicitation-interval 30
```

---

## 37.7 tunnel isatap robustness

Use the **tunnel isatap robustness** Global Configuration mode command to configure the number of DNS query/router solicitation refresh messages that the device sends. Use the **no** form of this command to restore the default configuration.

### Syntax

**tunnel isatap robustness** *number*

**no tunnel isatap robustness**

### Parameters

**number**—Specifies the number of DNS query/router solicitation refresh messages that the device sends. (Range: 1–20)

### Default Configuration

The default number of DNS query/router solicitation refresh messages that the device sends is 3.

### Command Mode

Global Configuration mode

### User Guidelines

The DNS query interval (after the ISATAP router IP address is known) is the Time-To-Live (TTL) that is received from the DNS, divided by (Robustness + 1).

The router solicitation interval (when there is an active ISATAP router) is the minimum-router-lifetime that is received from the ISATAP router, divided by (Robustness + 1).

### Example

The following example sets the number of DNS query/router solicitation refresh messages that the device sends to 5.

---

```
Console(config)# tunnel isatap robustness 5
```

---

## 37.8 show ipv6 tunnel

Use the **show ipv6 tunnel** EXEC mode command to display information on the ISATAP tunnel.

### Syntax

**show ipv6 tunnel**

### Command Mode

EXEC mode

### Example

The following example displays information on the ISATAP tunnel.

---

```
Console> show ipv6 tunnel

Tunnel 1
-----

Tunnel status                : DOWN
Tunnel protocol              : NONE
Tunnel Local address type    : auto
Tunnel Local Ipv4 address    : 0.0.0.0
Router DNS name              : ISATAP
Router IPv4 address          : 0.0.0.0
```

---

```
DNS Query interval          : 300 seconds
Min DNS Query interval      : 0 seconds
Router Solicitation interval : 10 seconds
Min Router Solicitation interval : 0 seconds
Robustness                   : 2
```

---

## 38 DHCP Relay Commands

---

### 38.1 ip dhcp relay enable (Global)

Use the **ip dhcp relay enable** Global Configuration mode command to enable the DHCP relay feature on the device. Use the **no** form of this command to disable the DHCP relay feature.

#### Syntax

**ip dhcp relay enable**

**no ip dhcp relay enable**

#### Parameters

N/A

#### Default Configuration

DHCP relay feature is disabled.

#### Command Mode

Global Configuration mode

#### Example

The following example enables the DHCP relay feature on the device.

---

```
Console(config)# ip dhcp relay enable
```

---

### 38.2 ip dhcp relay enable (Interface)

Use the **ip dhcp relay enable** Interface Configuration (VLAN, Ethernet, Port-channel) mode command to enable the DHCP relay feature on an interface. Use the **no** form of this command to disable the DHCP relay agent feature on an interface.

#### Syntax

**ip dhcp relay enable**

**no ip dhcp relay enable**

**Parameters**

N/A

**Default Configuration**

Disabled

**Command Mode**

Interface Configuration (VLAN, Ethernet, Port-channel) mode

**User Guidelines**

The operational status of DHCP Relay on an interface is active if one of the following conditions exist:

- DHCP Relay is globally enabled, and there is an IP address defined on the interface.

Or

- DHCP Relay is globally enabled, there is no IP address defined on the interface, the interface is a VLAN, and option 82 is enabled.

**Example**

The following example enables DHCP Relay on VLAN 21.

```
Console(config)# interface vlan 21
Console(config-if)# ip dhcp relay enable
```

**38.3 ip dhcp relay address**

Use the **ip dhcp relay address** Global Configuration mode command to define the DHCP servers available for the DHCP relay. Use the **no** form of this command to remove servers from the list.

**Syntax**

**ip dhcp relay address** *ip-address*

**no ip dhcp relay address** [*ip-address*]

### Parameters

**ip-address**—Specifies the DHCP server IP address. Up to 8 servers can be defined.

### Default Configuration

No server is defined.

### Command Mode

Global Configuration mode

### Example

The following example defines the DHCP server on the device.

---

```
Console(config)# ip dhcp relay address 176.16.1.1
```

---

## 38.4 show ip dhcp relay

Use the **show ip dhcp relay** EXEC mode command to display the DHCP relay information.

### Syntax

**show ip dhcp relay**

### Command Mode

EXEC mode

### Examples

**Example 1** - Option 82 is not supported:

---

```
console# show ip dhcp relay
DHCP relay is globally enabled
Option 82 is Disabled
Maximum number of supported VLANs without IP Address is 256
Number of DHCP Relays enabled on VLANs without IP Address is 0
DHCP relay is not configured on any port.
```

---

DHCP relay is not configured on any vlan.

No servers configured

---

### Example 2 - Option 82 is supported (disabled):

---

```
console# show ip dhcp relay
```

```
DHCP relay is globally disabled
```

```
Option 82 is disabled
```

```
Maximum number of supported VLANs without IP Address: 0
```

```
Number of DHCP Relays enabled on VLANs without IP Address: 4
```

```
DHCP relay is enabled on Ports: gi5,po3-4
```

```
Active:
```

```
Inactive: gi5, po3-4
```

```
DHCP relay is enabled on VLANs: 1, 2, 4, 5
```

```
Active:
```

```
Inactive: 1, 2, 4, 5
```

```
Servers: 1.1.1.1 , 2.2.2.2
```

---

### Example 3 - Option 82 is supported (enabled):

---

```
console# show ip dhcp relay
```

```
DHCP relay is globally enabled
```

```
Option 82 is enabled
```

```
Maximum number of supported VLANs without IP Address is 4
```

```
Number of DHCP Relays enabled on VLANs without IP Address: 2
```

```
DHCP relay is enabled on Ports: gi5,po3-4
```

```
Active: gi5
```

```
Inactive: po3-4
```

```
DHCP relay is enabled on VLANs: 1, 2, 4, 5
```

```
Active: 1, 2, 4, 5
```

```
Inactive:
```

---

---

Servers: 1.1.1.1 , 2.2.2.2



## 39 IP Routing Protocol-Independent Commands

### 39.1 ip route

Use the **ip route** Global Configuration mode command to configure static routes. Use the **no** form of this command to remove static routes.

#### Syntax

```
ip route prefix {mask | prefix-length} {{ip-address [metric distance]} | reject-route}  
no ip route prefix {mask | prefix-length} [ip-address]
```

#### Parameters

- **prefix**—Specifies the IP address that is the IP route prefix for the destination IP.
- **mask**—Specifies the network subnet mask of the IP address prefix.
- **prefix-length**—Specifies the number of bits that comprise the IP address prefix. The prefix length must be preceded by a forward slash (/). (Range: 0–32)
- **ip-address**—Specifies the IP address or IP alias of the next hop that can be used to reach the network.
- **metric distance**—Specifies an administrative distance. (Range: 1–255)
- **reject-route**—Stops routing to the destination network via all gateways.

#### Default Configuration

The default administrative distance is 1.

#### Command Mode

Global Configuration mode

#### User Guidelines

To use this command set the device in router mode with the command `set system mode router`.

Use the **no ip route** command without the *ip-address* parameter to remove all static routes to the given subnet.

Use the **no ip route** command with the *ip-address* parameter to remove only one static route to the given subnet via the given next hop.

### Examples

**Example 1** - The following example shows how to route packets for network 172.31.0.0 to a router at 172.31.6.6 using mask:

```
console(conf)#ip route 172.31.0.0 255.255.0.0 172.31.6.6 metric 2
```

**Example 2** - The following example shows how to route packets for network 172.31.0.0 to a router at 172.31.6.6 using prefix length:

```
console(conf)#ip route 172.31.0.0 /16 172.31.6.6 metric 2
```

**Example 3** - The following example shows how to reject packets for network 194.1.1.0:

```
console(conf)#ip route 194.1.1.0 255.255.255.0 reject-route
```

**Example 4** - The following example shows how to remove all static routes to network 194.1.1.0/24:

```
console(conf)#no ip route 194.1.1.0 /24
```

**Example 5** - The following example shows how to remove one static route to network 194.1.1.0/24 via 1.1.1.1:

```
console(conf)#no ip route 194.1.1.0 /24 1.1.1.1
```

## 39.2 show ip route

Use the **show ip route** EXEC mode command to display the current routing table state.

## Syntax

```
show ip route [connected / static / {address address [mask / prefix-length]
[longer-prefixes]]
```

## Parameters

- **connected**—Displays connected routing entries only.
- **static**—Displays static routing entries only.
- **address address**—Specifies the address for which routing information is displayed.
- **mask**—Specifies the network subnet mask of the IP address.
- **prefix-length**—Specifies the number of bits that comprise the IP address prefix. The prefix length must be preceded by a forward slash (/). (Range: 1–32)
- **longer-prefixes**—Specifies that the **address** and **mask** pair becomes a prefix and any routes that match that prefix are displayed.

## Command Mode

EXEC mode

## User Guidelines

To use this command set the device in router mode with the command [set system mode](#).

## Example

The following example displays the current routing table state.

---

```
Console> show ip route
console# show ip route
Maximum Parallel Paths: 1 (1 after reset)
IP Forwarding:          enabled
Codes: C - connected, S - static, D - DHCP
S 0.0.0.0/0             [gil] via 10.5.234.254 119:9:27  vlan 1
```

---

```
C 10.5.234.0/24      is directly connected          vlan 1
```

---

```
Console> show ip route address 172.1.1.0 255.255.255.0
```

```
Codes: C - connected, S - static, E - OSPF external, * - candidate default
```

```
S 172.1.1.0/24 [gi3] via 10.0.2.1, 17:12:19, gi1
```

---

```
Console> show ip route address 172.1.1.0 255.255.255.0 longer-prefixes
```

```
Codes: C - connected, S - static, E - OSPF external
```

```
S 172.1.1.0/24 [gi3] via 10.0.2.1, 17:12:19, gi1
```

```
S 172.1.1.1/32 [gi3] via 10.0.3.1, 19:51:18, gi1
```

---

The following table describes the significant fields shown in the display:

Field	Description
<b>O</b>	The protocol that derived the route.
<b>10.8.10/24</b>	The remote network address.
<b>[30/2000]</b>	The first number in the brackets is the administrative distance of the information source; the second number is the metric for the route.
<b>via 10.0.12</b>	The address of the next router to the remote network.
<b>00:39:08</b>	The last time the route was updated in hours:minutes:seconds.
<b>gi1</b>	The interface through which the specified network can be reached.

---

## 40 ACL Commands

---

### 40.1 ip access-list

Use the **ip access-list** Global Configuration mode command to name an IPv4 access list (ACL) and to place the device in IPv4 Access List Configuration mode. All commands after this command refer to this ACL. The rules (ACEs) for this ACL are defined in the **permit (IP)** and **deny (IP)** commands. The **service-acl** command is used to attach this ACL to an interface.

Use the **no** form of this command to remove the access list.

#### Syntax

**ip access-list extended** *acl-name*

**no ip access-list extended** *acl-name*

#### Parameters

- **acl-name**—Name of the IPv4 access list.
- **acl-name**—0–32 characters. (Use "" for empty string)

#### Default Configuration

No IPv4 access list is defined.

#### Command Mode

Global Configuration mode

#### User Guidelines

An IPv4 ACL is defined by a unique name. IPv4 ACL, IPv6 ACL, MAC ACL or policy maps cannot have the same name.

#### Example

---

```
console(config)# ip access-list extended server
console(config-ip-acl)#
```

## 40.2 permit ( IP )

Use the **permit** IP Access-list Configuration mode command to set permit conditions for an IPv4 access list (ACL). Permit conditions are also known as access control entries (ACEs).

### Syntax

```
permit protocol {any | source source-wildcard} {any | destination destination-wildcard} [dscp number | precedence number]
```

```
permit icmp {any | source source-wildcard} {any | destination destination-wildcard} [any | icmp-type] [any | icmp-code]] [dscp number | precedence number]
```

```
permit igmp {any | source source-wildcard} {any | destination destination-wildcard}] [igmp-type] [dscp number | precedence number]
```

```
permit tcp {any | source source-wildcard} {any | source-port/port-range} {any | destination destination-wildcard} {any | destination-port/port-range} [dscp number | precedence number] [match-all list-of-flags]
```

```
permit udp {any | source source-wildcard} {any | source-port/port-range} {any | destination destination-wildcard} {any | destination-port/port-range} [dscp number | precedence number]
```

### Parameters

- **permit** *protocol*—The name or the number of an IP protocol. Available protocol names are: icmp, igmp, ip, tcp, egp, igp, udp, hmp, rdp, idpr, ipv6, ipv6:rout, ipv6:frag, idrp, rsvp, gre, esp, ah, ipv6:icmp, eigrp, ospf, ipinip, pim, l2tp, isis. To match any protocol, use the **ip** keyword.(Range: 0–255)
- **source**—Source IP address of the packet.
- **source-wildcard**—Wildcard bits to be applied to the source IP address. Use ones in the bit position that you want to be ignored.
- **destination**—Destination IP address of the packet.
- **destination-wildcard**—Wildcard bits to be applied to the destination IP address. Use ones in the bit position that you want to be ignored.
- **dscp** *number*—Specifies the DSCP value.
- **precedence** *number*—Specifies the IP precedence value.
- **icmp-type**—Specifies an ICMP message type for filtering ICMP packets. Enter a number or one of the following values: echo-reply, destination-unreachable, source-quench, redirect, alternate-host-address,

echo-request, router-advertisement, router-solicitation, time-exceeded, parameter-problem, timestamp, timestamp-reply, information-request, information-reply, address-mask-request, address-mask-reply, traceroute, datagram-conversion-error, mobile-host-redirect, mobile-registration-request, mobile-registration-reply, domain-name-request, domain-name-reply, skip, photuris. (Range: 0–255)

- **icmp-code**—Specifies an ICMP message code for filtering ICMP packets. (Range: 0–255)
- **igmp-type**—IGMP packets can be filtered by IGMP message type. Enter a number or one of the following values: host-query, host-report, dvmrp, pim, cisco-trace, host-report-v2, host-leave-v2, host-report-v3. (Range: 0–255)
- **destination-port**—Specifies the UDP/TCP destination port. You can enter range of ports by using hyphen. E.g. 20 - 21. For TCP enter a number or one of the following values: bgp (179), chargen (19), daytime (13), discard (9), domain (53), drip (3949), echo (7), finger (79), ftp (21), ftp-data (20), gopher (70), hostname (42), irc (194), klogin (543), kshell (544), lpd (515), nntp (119), pop2 (109), pop3 (110), smtp (25), sunrpc (1110), syslog (514), tacacs-ds (49), talk (517), telnet (23), time (37), uucp (117), whois (43), www (80). For UDP enter a number or one of the following values: biff (512), bootpc (68), bootps (67), discard (9), dnsix (90), domain (53), echo (7), mobile-ip (434), nameserver (42), netbios-dgm (138), netbios-ns (137), on500-isakmp (4500), ntp (123), rip (520), snmp (161), snmptrap (162), sunrpc (111), syslog (514), tacacs-ds (49), talk (517), tftp (69), time (37), who (513), xdmcp (177). (Range: 0–65535).
- **source-port**—Specifies the UDP/TCP source port. Predefined port names are defined in the destination-port parameter. (Range: 0–65535)
- **match-all *list-of-flags***—List of TCP flags that should occur. If a flag should be set, it is prefixed by “+”. If a flag should be unset, it is prefixed by “-”. Available options are +urg, +ack, +psh, +rst, +syn, +fin, -urg, -ack, -psh, -rst, -syn and -fin. The flags are concatenated to a one string. For example: +fin-ack.

### Default Configuration

No IPv4 access list is defined.

### Command Mode

IP Access-list Configuration mode

### User Guidelines

After an ACE is added to an access control list, an implicit **deny any** condition exists at the end of the list. That is, if there are no matches, the packets are denied. However, before the first ACE is added, the list permits all packets up to #ASIC-specific ranges for TCP and up to #ASIC-specific ranges for UDP. If a range of ports is used for source port in an ACE, it is not counted again, if it is also used for a source port in another ACE. If a range of ports is used for the destination port in an ACE, it is not counted again if it is also used for destination port in another ACE.

If a range of ports is used for source port it is counted again if it is also used for destination port.

### Example

```
console(config)# ip access-list extended server
console(config-ip-al)# permit ip 176.212.0.0 00.255.255
```

## 40.3 deny ( IP )

Use the **deny IP Access-list Configuration mode** command to set deny conditions for IPv4 access list. Deny conditions are also known as access control entries (ACEs).

### Syntax

**deny protocol** {*any* | *source source-wildcard*} {*any* | *destination destination-wildcard*} [*dscp number* | *precedence number*]

**deny icmp** {*any* | *source source-wildcard*} {*any* | *destination destination-wildcard*} [*any* | *icmp-type*] [*any* | *icmp-code*]] [*dscp number* | *precedence number*]

**deny igmp** {*any* | *source source-wildcard*} {*any* | *destination destination-wildcard*}[*igmp-type*] [*dscp number* | *precedence number*]

**deny tcp** {*any* | *source source-wildcard*} {*any*|*source-port/port-range*}{*any* | *destination destination-wildcard*} {*any*|*destination-port/port-range*} [*dscp number* | *precedence number*] [*match-all list-of-flags*]

**deny udp** {*any* | *source source-wildcard*} {*any*|*source-port/port-range*} {*any* | *destination destination-wildcard*} {*any*|*destination-port/port-range*} [*dscp number* | *precedence number*]



## Parameters

- **protocol**—The name or the number of an IP protocol. Available protocol names: icmp, igmp, ip, tcp, egp, igp, udp, hmp, rdp, idpr, ipv6, ipv6:rout, ipv6:frag, idrp, rsvp, gre, esp, ah, ipv6:icmp, eigrp, ospf, ipinip, pim, l2tp, isis. To match any protocol, use the ip keyword. (Range: 0–255)
- **source**—Source IP address of the packet.
- **source-wildcard**—Wildcard bits to be applied to the source IP address. Use 1s in the bit position that you want to be ignored.
- **destination**—Destination IP address of the packet.
- **destination-wildcard**—Wildcard bits to be applied to the destination IP address. Use 1s in the bit position that you want to be ignored.
- **dscp number**—Specifies the DSCP value.
- **precedence number**—Specifies the IP precedence value.
- **icmp-type**—Specifies an ICMP message type for filtering ICMP packets. Enter a number or one of the following values: echo-reply, destination-unreachable, source-quench, redirect, alternate-host-address, echo-request, router-advertisement, router-solicitation, time-exceeded, parameter-problem, timestamp, timestamp-reply, information-request, information-reply, address-mask-request, address-mask-reply, traceroute, datagram-conversion-error, mobile-host-redirect, mobile-registration-request, mobile-registration-reply, domain-name-request, domain-name-reply, skip, photuris. (Range: 0–255)
- **icmp-code**—Specifies an ICMP message code for filtering ICMP packets. (Range: 0–255)
- **igmp-type**—IGMP packets can be filtered by IGMP message type. Enter a number or one of the following values: host-query, host-report, dvmrp, pim, cisco-trace, host-report-v2, host-leave-v2, host-report-v3. (Range: 0–255)
- **destination-port**—Specifies the UDP/TCP destination port. You can enter range of ports by using hyphen. E.g. 20 - 21. For TCP enter a number or one of the following values: bgp (179), chargen (19), daytime (13), discard (9), domain (53), drip (3949), echo (7), finger (79), ftp (21), ftp-data (20), gopher (70), hostname (42), irc (194), klogin (543), kshell (544), lpd (515), nntp (119), pop2 (109), pop3 (110), smtp (25), sunrpc (1110), syslog (514), tacacs-ds (49), talk (517), telnet (23), time (37), uucp (117), whois (43), www (80). For UDP enter a number or one of the following values: biff (512), bootpc (68), bootps (67), discard (9), dnsix (90), domain (53), echo (7), mobile-ip (434),

nameserver (42), netbios-dgm (138), netbios-ns (137), non500-isakmp (4500), ntp (123), rip (520), snmp (161), snmptrap (162), sunrpc (111), syslog (514), tacacs-ds (49), talk (517), tftp (69), time (37), who (513), xdmcp (177). (Range: 0–65535)

- **source-port**—Specifies the UDP/TCP source port. Predefined port names are defined in the destination-port parameter. (Range: 0–65535)
- **match-all *list-of-flags***—List of TCP flags that should occur. If a flag should be set it is prefixed by “+”. If a flag should be unset it is prefixed by “-”. Available options are +urg, +ack, +psh, +rst, +syn, +fin, -urg, -ack, -psh, -rst, -syn and -fin. The flags are concatenated to a one string. For example: +fin-ack.
- **disable-port**—The Ethernet interface is disabled if the condition is matched.
- **log-input**—Specifies sending an informational syslog message about the packet that matches the entry. Because forwarding is done in hardware and logging is done in software, if a large number of packets match a deny ACE containing a log-input keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

### Default Configuration

No IPv4 access list is defined.

### Command Mode

IP Access-list Configuration mode

### User Guidelines

After an ACE is added to an access control list, an implicit **deny any any** condition exists at the end of the list. That is, if there are no matches, the packets are denied. However, before the first ACE is added, the list permits all packets.

The number of TCP/UDP ranges that can be defined in ACLs is limited. You can define up to #ASIC-specific ranges for TCP and up to #ASIC-specific ranges for UDP. If a range of ports is used for a source port in ACE it is not counted again if it is also used for source port in another ACE. If a range of ports is used for destination port in ACE it is not counted again if it is also used for destination port in another ACE.

If a range of ports is used for source port, it is counted again if it is also used for destination port.

### Example

---

```
console(config)# ip access-list extended server
console(config-ip-af)# deny ip 176.212.0.0 0.255.255
```

---

## 40.4 ipv6 access-list

Use the **ipv6 access-list** Global Configuration mode command to define an IPv6 access list (ACL) and to place the device in IPv6 Access List Configuration mode. All commands after this command refer to this ACL. The rules (ACEs) for this ACL are defined in the [permit \(IPv6\)](#) and [deny \(IPv6\)](#) commands. The [service-acl](#) command is used to attach this ACL to an interface.

Use the **no** form of this command to remove the access list.

### Syntax

```
ipv6 access-list [acl-name]
```

```
no ipv6 access-list [acl-name]
```

### Parameters

- **acl-name**—Name of the IPv6 access list. Range 0-32 characters (use "" for empty string).

### Default Configuration

No IPv6 access list is defined.

### Command Mode

Global Configuration mode

### User Guidelines

IPv6 ACL is defined by a unique name. IPv4 ACL, IPv6 ACL, MAC ACL or policy maps cannot have the same name.

Every IPv6 ACL has an implicit **permit icmp any any nd-ns any**, **permit icmp any any nd-na any**, and **deny ipv6 any any** statements as its last match conditions. (The former two match conditions allow for ICMPv6 neighbor discovery.)

The IPv6 neighbor discovery process uses the IPv6 network layer service, therefore, by default, IPv6 ACLs implicitly allow IPv6 neighbor discovery packets to be sent and received on an interface. In IPv4, the Address Resolution Protocol (ARP), which is equivalent to the IPv6 neighbor discovery process, uses a

separate data link layer protocol; therefore, by default, IPv4 ACLs implicitly allow ARP packets to be sent and received on an interface.

### Example

```
Switch (config)# ipv6 access-list acl1
Switch(config-ipv6-acl)# permit tcp 2001:0DB8:0300:0201::/64 any any 80
```

## 40.5 permit ( IPv6 )

Use the **permit** command in IPv6 Access-list Configuration mode to set permit conditions (ACEs) for IPv6 ACLs.

### Syntax

**permit** *protocol* {*any* | {*source-prefix/length*}}{*any* | *destination-prefix/length*} [*dscp number* | *precedence number*]

**permit icmp** {*any* | {*source-prefix/length*}}{*any* | *destination-prefix/length*} {*any*|*icmp-type*} {*any*|*icmp-code*} [*dscp number* | *precedence number*]

**permit tcp** {*any* | {*source-prefix/length*}} {*any* | *source-port/port-range*}}{*any* | *destination-prefix/length*} {*any*| *destination-port/port-range*} [*dscp number* | *precedence number*] [*match-all list-of-flags*]

**permit udp** {*any* | {*source-prefix/length*}} {*any* | *source-port/port-range*}}{*any* | *destination-prefix/length*} {*any*| *destination-port/port-range*} [*dscp number* | *precedence number*]

### Parameters

- **protocol**—The name or the number of an IP protocol. Available protocol names are: icmp (58), tcp (6) and udp (17). To match any protocol, use the ipv6 keyword. (Range: 0–255)
- **source-prefix/length**—The source IPv6 network or class of networks about which to set permit conditions. This argument must be in the form documented in RFC 3513 where the address is specified in hexadecimal using 16-bit values between colons.
- **destination-prefix/length**—The destination IPv6 network or class of networks about which to set permit conditions. This argument must be in the form documented in RFC 3513 where the address is specified in hexadecimal using 16-bit values between colons.

- **dscp *number***—Specifies the DSCP value. (Range: 0–63)
- **precedence *number***—Specifies the IP precedence value.
- **icmp-type**—Specifies an ICMP message type for filtering ICMP packets. Enter a number or one of the following values: destination-unreachable (1), packet-too-big (2), time-exceeded (3), parameter-problem (4), echo-request (128), echo-reply (129), mld-query (130), mld-report (131), mldv2-report (143), mld-done (132), router-solicitation (133), router-advertisement (134), nd-ns (135), nd-na (136). (Range: 0–255)
- **icmp-code**—Specifies an ICMP message code for filtering ICMP packets. (Range: 0–255)
- **destination-port**—Specifies the UDP/TCP destination port. You can enter a range of ports by using a hyphen. E.g. 20 - 21. For TCP enter a number or one of the following values: bgp (179), chargen (19), daytime (13), discard (9), domain (53), drip (3949), echo (7), finger (79), ftp (21), ftp-data (20), gopher (70), hostname (42), irc (194), klogin (543), kshell (544), lpd (515), nntp (119), pop2 (109), pop3 (110), smtp (25), sunrpc (1110), syslog (514), tacacs-ds (49), talk (517), telnet (23), time (37), uucp (117), whois (43), www (80). For UDP enter a number or one of the following values: biff (512), bootpc (68), bootps (67), discard (9), dnsix (90), domain (53), echo (7), mobile-ip (434), nameserver (42), netbios-dgm (138), netbios-ns (137), non500-isakmp (4500), ntp (123), rip (520), snmp (161), snmptrap (162), sunrpc (111), syslog (514), tacacs (49), talk (517), tftp (69), time (37), who (513), xdmcp (177). (Range: 0–65535)
- **source-port**—Specifies the UDP/TCP source port. Predefined port names are defined in the destination-port parameter. (Range: 0–65535)
- **match-all *list-of-flag***—List of TCP flags that should occur. If a flag should be set it is prefixed by “+”. If a flag should be unset it is prefixed by “-”. Available options are +urg, +ack, +psh, +rst, +syn, +fin, -urg, -ack, -psh, -rst, -syn and -fin. The flags are concatenated to a one string. For example: +fin-ack.

### Default Configuration

No IPv6 access list is defined.

### Command Mode

IPv6 Access-list Configuration mode

### User Guidelines

The number of TCP/UDP ranges that can be defined in ACLs is limited. You can define up to #ASIC-specific ranges for TCP and up to #ASIC-specific ranges for UDP. If a range of ports is used for a source port in ACE, it is not counted again if it is also used for a source port in another ACE. If a range of ports is used for destination port in ACE it is not counted again if it is also used for destination port in another ACE.

If a range of ports is used for source port it is counted again if it is also used for destination port.

### Example

This example defines an ACL by the name of server and enters a rule (ACE) for tcp packets.

```
console(config)# ipv6 access-list server
console(config-ipv6-al)# permit tcp 3001::2/64 any any 80
```

## 40.6 deny (IPv6)

Use the **deny** command in IPv6 Access List Configuration mode to set permit conditions (ACEs) for IPv6 ACLs.

### Syntax

**deny protocol** {*any* | {*source-prefix/length*}}{*any* | *destination-prefix/length*} [*dscp number* | *precedence number*] [*disable-port* | *log-input*]

**deny icmp** {*any* | {*source-prefix/length*}}{*any* | *destination-prefix/length*} {*any/icmp-type*} {*any/icmp-code*} [*dscp number* | *precedence number*] [*disable-port* | *log-input*]

**deny tcp** {*any* | {*source-prefix/length*} {*any* | *source-port/port-range*}}{*any* | *destination-prefix/length*} {*any* | *destination-port/port-range*} [*dscp number* | *precedence number*] [*match-all list-of-flags*] [*disable-port* | *log-input*]

**deny udp** {*any* | {*source-prefix/length*}} {*any* | *source-port/port-range*}}{*any* | *destination-prefix/length*} {*any* | *destination-port/port-range*} [*dscp number* | *precedence number*] [*disable-port* | *log-input*]

## Parameters

- **protocol**—The name or the number of an IP protocol. Available protocol names are: icmp (58), tcp (6) and udp (17). To match any protocol, use the ipv6 keyword. (Range: 0–255)
- **source-prefix/length**—The source IPv6 network or class of networks about which to set permit conditions. This argument must be in the format documented in RFC 3513 where the address is specified in hexadecimal using 16-bit values between colons.
- **destination-prefix/length**—The destination IPv6 network or class of networks about which to set permit conditions. This argument must be in the format documented in RFC 3513 where the address is specified in hexadecimal using 16-bit values between colons.
- **dscp number**—Specifies the DSCP value. (Range: 0–63)
- **precedence number**—Specifies the IP precedence value.
- **icmp-type**—Specifies an ICMP message type for filtering ICMP packets. Enter a number or one of the following values: destination-unreachable (1), packet-too-big (2), time-exceeded (3), parameter-problem (4), echo-request (128), echo-reply (129), mld-query (130), mld-report (131), mldv2-report (143), mld-done (132), router-solicitation (133), router-advertisement (134), nd-ns (135), nd-na (136). (Range: 0–255)
- **icmp-code**—Specifies an ICMP message code for filtering ICMP packets. (Range: 0–255)
- **destination-port**—Specifies the UDP/TCP destination port. You can enter a range of ports by using a hyphen. E.g. 20 - 21. For TCP enter a number or one of the following values: bgp (179), chargen (19), daytime (13), discard (9), domain (53), drip (3949), echo (7), finger (79), ftp (21), ftp-data (20), gopher (70), hostname (42), irc (194), klogin (543), kshell (544), lpd (515), nntp (119), pop2 (109), pop3 (110), smtp (25), sunrpc (1110), syslog (514), tacacs-ds (49), talk (517), telnet (23), time (37), uucp (117), whois (43), www (80). For UDP enter a number or one of the following values: biff (512), bootpc (68), bootps (67), discard (9), dnsix (90), domain (53), echo (7), mobile-ip (434), nameserver (42), netbios-dgm (138), netbios-ns (137), non500-isakmp (4500), ntp (123), rip (520), snmp (161), snmptrap (162), sunrpc (111), syslog (514), tacacs (49), talk (517), tftp (69), time (37), who (513), xdmcp (177). (Range: 0–65535)
- **source-port**—Specifies the UDP/TCP source port. Predefined port names are defined in the destination-port parameter. (Range: 0–65535)

- **match-all *list-of-flags***—List of TCP flags that should occur. If a flag should be set it is prefixed by “+”. If a flag should be unset it is prefixed by “-”. Available options are +urg, +ack, +psh, +rst, +syn, +fin, -urg, -ack, -psh, -rst, -syn and -fin. The flags are concatenated to a one string. For example: +fin-ack.
- **disable-port**—The Ethernet interface is disabled if the condition is matched.
- **log-input**—Specifies to send an informational syslog message about the packet that matches the entry. Because forwarding is done in hardware and logging is done in software, if a large number of packets match a deny ACE containing a log-input keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

### Default Configuration

No IPv6 access list is defined.

### Command Mode

IPv6 Access-list Configuration mode

### User Guidelines

The number of TCP/UDP ranges that can be defined in ACLs is limited. You can define up to #ASIC-specific ranges for TCP and up to #ASIC-specific ranges for UDP. If a range of ports is used for source port in ACE it is not counted again if it is also used for source port in another ACE. If a range of ports is used for a destination port in ACE it is not counted again if it is also used for a destination port in another ACE.

If a range of ports is used for source port it is counted again if it is also used for destination port.

### Example

```
console(config)# ipv6 access-list server
console(config-ipv6-al)# deny tcp 3001::2/64 any any 80
```

## 40.7 mac access-list

Use the **mac access-list** Global Configuration mode command to define a Layer 2 access list (ACL) based on source MAC address filtering and to place the device in MAC Access List Configuration mode. All commands after this command refer to this ACL. The rules (ACEs) for this ACL are defined in the [permit \(MAC\)](#) and



**deny (MAC)** commands. The **service-acl** command is used to attach this ACL to an interface.

Use the **no** form of this command to remove the access list.

### Syntax

**mac access-list extended** *acl-name*

**no mac access-list extended** *acl-name*

### Parameters

**acl-name**—Specifies the name of the MAC ACL (range: 0–32 characters - use "" for empty string).

### Default Configuration

No MAC access list is defined.

### Command Mode

Global Configuration mode

### User Guidelines

A MAC ACL is defined by a unique name. IPv4 ACL, IPv6 ACL, MAC ACL or policy maps cannot have the same name.

### Example

```
console(config)# mac access-list extended server1
console(config-mac-acl)# permit 00:00:00:00:00:01 00:00:00:00:00:ff any
```

---

## 40.8 permit ( MAC )

Use the **permit** command in MAC Access List Configuration mode to set permit conditions (ACEs) for a MAC ACL.

### Syntax

**permit** *{any | source source-wildcard} {any | destination destination-wildcard}*  
*[eth-type 0 | aarp | amber | dec-spanning | decnet-iv | diagnostic | dsm | etype-6000]*  
*[vlan vlan-id] [cos cos cos-wildcard]*

### Parameters

- **source**—Source MAC address of the packet.
- **source-wildcard**—Wildcard bits to be applied to the source MAC address. Use 1s in the bit position that you want to be ignored.
- **destination**—Destination MAC address of the packet.
- **destination-wildcard**—Wildcard bits to be applied to the destination MAC address. Use 1s in the bit position that you want to be ignored.
- **eth-type**—The Ethernet type in hexadecimal format of the packet.
- **vlan-id**—The VLAN ID of the packet. (Range: 1–4094)
- **cos**—The Class of Service of the packet. (Range: 0–7)
- **cos-wildcard**—Wildcard bits to be applied to the CoS.

### Default Configuration

No MAC access list is defined.

### Command Mode

MAC Access-list Configuration mode

### User Guidelines

After an access control entry (ACE) is added to an access control list, an implicit **deny any any** condition exists at the end of the list. That is, if there are no matches, the packets are denied. However, before the first ACE is added, the list permits all packets.

### Example

```
console(config)# mac access-list extended server1
console(config-mac-al)# permit 00:00:00:00:00:01 00:00:00:00:00:ff any
```

---

## 40.9 deny (MAC)

Use the **deny** command in MAC Access List Configuration mode to set deny conditions (ACEs) for a MAC ACL.

## Syntax

```
deny {any / source source-wildcard} {any / destination destination-wildcard}  
[[eth-type 0] aarp / amber / dec-spanning / decnet-iv / diagnostic / dsm /  
etype-6000] [vlan vlan-id] [cos cos cos-wildcard] [disable-port / log-input]
```

## Parameters

- **source**—Source MAC address of the packet.
- **source-wildcard**—Wildcard bits to be applied to the source MAC address. Use ones in the bit position that you want to be ignored.
- **destination**—Destination MAC address of the packet.
- **destination-wildcard**—Wildcard bits to be applied to the destination MAC address. Use 1s in the bit position that you want to be ignored.
- **eth-type**—The Ethernet type in hexadecimal format of the packet.
- **vlan-id**—The VLAN ID of the packet. (Range: 1–4094)
- **cos**—The Class of Service of the packet.(Range: 0–7)
- **cos-wildcard**—Wildcard bits to be applied to the CoS.
- **disable-port**—The Ethernet interface is disabled if the condition is matched.
- **log-input**—Sends an informational syslog message about the packet that matches the entry. Because forwarding is done in hardware and logging is done in software, if a large number of packets match a deny ACE containing a log-input keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

## Default Configuration

No MAC access list is defined.

## Command Mode

MAC Access-list Configuration mode

## User Guidelines

After an access control entry (ACE) is added to an access control list, an implicit **deny any any** condition exists at the end of the list. That is, if there are no matches, the packets are denied. However, before the first ACE is added, the list permits all packets.

---

### Example

```
console(config)# mac access-list extended server1
console(config-mac-al)# deny 00:00:00:00:00:01 00:00:00:00:00:ff any
```

---

## 40.10 service-acl

use the **service-acl** command in interface Configuration mode to bind an access list(s) (ACL) to an interface.

Use the **no** form of this command to remove all ACLs from the interface.

### Syntax

```
service-acl input acl-name1 [acl-name2]
```

```
no service-acl input
```

### Parameters

**acl-name**—Specifies an ACL to apply to the interface. See the user guidelines. (Range: 0–32 characters). Use "" for empty string).

### Default Configuration

No ACL is assigned.

### Command Mode

Interface Configuration (Ethernet, Port-Channel) mode.

### User Guidelines

The following rules govern when ACLs can be bound or unbound from an interface:

- IPv4 ACLs and IPv6 ACLs can be bound together to an interface.
- A MAC ACL cannot be bound on an interface which already has an IPv4 ACL or IPv6 ACL bound to it.
- Two ACLs of the same type cannot be bound to a port.
- An ACL cannot be bound to a port that is already bound to an ACL, without first removing the current ACL. Both ACLs must be mentioned at the same time in this command.

---

### Example

```
console(config)# mac access-list extended server-acl
console(config-mac-acl)# permit 00:00:00:00:00:01 00:00:00:00:00:ff any
console(config-mac-acl)# exit
console(config)# interface gil
console(config-if)# service-acl input server-acl
```

---

## 40.11 absolute

Use the **absolute** Time-range Configuration mode command to specify an absolute time when a time range is in effect. Use the **no** form of this command to remove the time limitation.

### Syntax

**absolute** *start* *hh:mm day month year*

**no absolute** *start*

**absolute** *end* *hh:mm day month year*

**no absolute** *end*

### Parameters

- **start**—Absolute time and date that the permit or deny statement of the associated function going into effect. If no start time and date are specified, the function is in effect immediately.
- **end**—Absolute time and date that the permit or deny statement of the associated function is no longer in effect. If no end time and date are specified, the function is in effect indefinitely.
- **hh:mm**—Time in hours (military format) and minutes (Range: 0–23, mm: 0–5)
- **day**—Day (by date) in the month. (Range: 1–31)
- **month**—Month (first three letters by name). (Range: Jan...Dec)
- **year**—Year (no abbreviation) (Range: 2000–2097)

### Default Configuration

There is no absolute time when the time range is in effect.

## Command Mode

Time-range Configuration mode

## Example

```
Console (config)# time-range  
Console (config-time-range)# absolute start 12:00 1 jan 2005  
Console (config-time-range)# absolute end 12:00 31 dec 2005
```

## 40.12 periodic

Use the **periodic** Time-range Configuration mode command to specify a recurring (weekly) time range for functions that support the time-range feature. Use the **no** form of this command to remove the time limitation.

### Syntax

**periodic** *day-of-the-week hh:mm to day-of-the-week hh:mm*

**no periodic** *day-of-the-week hh:mm to day-of-the-week hh:mm*

**periodic list** *hh:mm to hh:mm day-of-the-week1 [day-of-the-week2...  
day-of-the-week7]*

**no periodic list** *hh:mm to hh:mm day-of-the-week1 [day-of-the-week2...  
day-of-the-week7]*

**periodic list** *hh:mm to hh:mm all*

**no periodic list** *all hh:mm to hh:mm all*

### Parameters

- **day-of-the-week**—The starting day that the associated time range is in effect. The second occurrence is the ending day the associated statement is in effect. The second occurrence can be the following week (see description in the User Guidelines). Possible values are: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday.
- **hh:mm**—The first occurrence of this argument is the starting hours:minutes (military format) that the associated time range is in effect. The second occurrence is the ending hours:minutes (military format) the associated statement is in effect. The second occurrence can be at the following day (see description in the User Guidelines). (Range: 0–23, mm: 0–59)

- **list day-of-the-week1**—Specifies a list of days that the time range is in effect.

### Default Configuration

There is no periodic time when the time range is in effect.

### Command Mode

Time-range Configuration mode

### User Guidelines

The second occurrence of the day can be at the following week, e.g. Thursday–Monday means that the time range is effective on Thursday, Friday, Saturday, Sunday, and Monday.

The second occurrence of the time can be on the following day, e.g. “22:00–2:00”.

### Example

---

```
Console (config)# time-range
Console (config-time-range)# periodic Monday 12:00 to Wednesday 12:00
```

---

## 40.13 show time-range

Use the **show time-range EXEC** command to display the time range configuration.

### Syntax

**show time-range** *time-range-name*

### Parameters

**time-range-name**—Specifies the name of an existing time range.

### Command Mode

EXEC mode

### Example

---

```
Console# show time-range
http-allowed
```

```
-----  
absolute start 12:00 1 jan 2005  
absolute end 12:00 31 dec 2005  
periodic monday 8:00 to friday 20:00
```

---

## 40.14 show access-lists

Use the **show access-lists** Privileged EXEC mode command to display access control lists (ACLs) configured on the switch.

### Syntax

**show access-lists** [*name*]

**show access-lists**

### Parameters

- **name**—Specifies the name of the ACL.

### Command Mode

Privileged EXEC mode

### Example

---

```
Console#show access-lists  
Standard IP access list 1  
deny any  
Standard IP access list 2  
deny 192.168.0.0, wildcard bits 0.0.0.255  
permit any  
Standard IP access list 3  
deny 0.0.0.0  
deny 192.168.0.1, wildcard bits 0.0.0.255  
permit any  
Standard IP access list ACL1  
permit 0.0.0.0
```



```
permit 192.168.0.2, wildcard bits 0.0.0.255
```

```
Extended IP access list ACL1
```

```
permit 234 172.30.40.1 0.0.0.0 any
```

```
permit 234 172.30.8.8 0.0.0.0 any
```

```
Extended IP access list ACL2
```

```
permit 234 172.30.19.1 0.0.0.255 any
```

```
permit 234 172.30.23.8 0.0.0.255 any
```

```
Console#show access-lists time-range-active
```

```
Extended IP access list ACL1
```

```
permit 234 172.30.40.1 0.0.0.0 any
```

```
permit 234 172.30.8.8 0.0.0.0 any
```

```
Extended IP access list ACL2
```

```
permit 234 172.30.19.1 0.0.0.255 any time-range weekdays
```

---

```
Console#show access-lists ACL1
```

```
Standard IP access list ACL1
```

```
permit 0.0.0.0
```

```
permit 192.168.0.2, wildcard bits 0.0.0.255
```

```
Extended IP access list ACL1
```

```
permit 234 172.30.40.1 0.0.0.0 any
```

```
permit 234 172.30.8.8 0.0.0.0 any
```

---

## 40.15 show interfaces access-lists

Use the **show interfaces access-lists** Privileged EXEC mode command to display access lists (ACLs) applied on interfaces.

### Syntax

**show interfaces access-lists** [*interface-id*]

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port, port-channel or VLAN.

### Command Mode

Privileged EXEC mode

### Example

---

```
Console# show interfaces access-lists
Interface      Input ACL
-----
gi1            ACL1
gi2            ACL3
gi3            blockedp, blockvtp
```

---

## 40.16 clear access-lists counters

Use the **clear access-lists counters** Privileged EXEC mode command to clear access-lists (ACLs) counters.

### Syntax

**clear access-lists counters** [*interface-id*]

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or port-channel.

### Command Mode

Privileged EXEC mode

### Example

```
console# clear access-lists counters gil
```

## 40.17 show interfaces access-lists counters

Use the **show interfaces access-lists counters** Privileged EXEC mode command to display Access List (ACLs) counters.

### Syntax

```
show interfaces access-lists counters [interface-id / port-channel-number]
```

### Parameters

**interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or port-channel.

### Command Mode

Privileged EXEC mode

### User Guidelines

The deny ACE hits count includes only ACEs with the log-input keyword.

Because forwarding is done in hardware and counting is done in software, if a large number of packets match a deny ACE containing a log-input keyword, the software might not be able to match the hardware processing rate, and not all packets are counted.

### Example

```
console# show interfaces access-lists counters
```

```
Interface                deny ACE hits
-----                -
gil                       79
```

---

```
gi2          9
```

```
gi3          0
```

```
Number of hits that were counted in global counter (due to lack of resources) =19
```

# 41 Quality of Service (QoS) Commands

## 41.1 qos

Use the **qos** Global Configuration mode command to enable Quality of Service (QoS) on the device and set the QoS mode. Use the **no** form of this command to disable QoS on the device

### Syntax

```
qos [basic | advanced [ports-not-trusted / ports-trusted]]
```

```
no qos
```

### Parameters

- **basic**—QoS basic mode. If no option is specified, the QoS mode defaults to the basic mode.
- **advanced**—Specifies the QoS advanced mode, which enables the full range of QoS configuration.
- **ports-not-trusted**—Relevant for advanced mode only. Indicates that packets, which are not classified by policy map rules to a QoS action, are mapped to egress queue 0. This is the default setting in advanced mode.
- **ports-trusted**—Relevant for advanced mode only. Indicates that packets, which are not classified by policy map rules to a QoS action, are mapped to an egress queue based on the packet's fields. Use the [qos advanced-mode trust](#) command to specify the trust mode.

### Default Configuration

If **qos** is entered without any keywords, the QoS **basic** mode is enabled.

If **qos advanced** is entered without a keyword, the default is **ports-not-trusted**.

### Command Mode

Global Configuration mode

### Example

**Example 1-** The following example enables the QoS basic mode on the device.

```
Console(config)# qos
```

---

**Example 2** - The following example enables the QoS advanced mode on the device with the **ports-not-trusted** option.

---

```
Console(config)# qos advanced
```

---

## 41.2 qos advanced-mode trust

Use the **qos advanced-mode trust** global configuration command to configure the trust mode in advanced mode. Use the **no** form of this command to return to default.

### Syntax

```
qos advanced-mode trust {cos | dscp | cos-dscp}
```

```
no qos advanced-mode trust
```

### Parameters

**cos**—Classifies ingress packets with the packet CoS values. For untagged packets, the port default CoS is used.

**dscp**—Classifies ingress packets with the packet DSCP values.

**cos-dscp**—Classifies ingress packets with the packet DSCP values for IP packets. For other packet types, use the packet CoS values.

### Default Configuration

```
cos-dscp
```

### Command Mode

Global configuration

### User Guidelines

The configuration is relevant for advanced mode in the following cases:

- **ports-not-trusted mode:** For packets that are classified to the QoS action trust.

- **ports-trusted mode:** For packets that are not classified by to any QoS action or classified to the QoS action trust.

### Example

The following example sets **cos** as the trust mode for QoS on the device.

---

```
Console(config)# qos advanced-mode trust cos
```

---

## 41.3 show qos

Use the **show qos** EXEC mode command to display the QoS information for the device. The trust mode is displayed for the QoS basic mode.

### Syntax

**show qos**

### Parameters

N/A

### Default Configuration

Disabled Command Mode

### Command Mode

EXEC mode

### User Guidelines

Trust mode is displayed if QoS is enabled in basic mode.

### Examples

**Example 1** - The following example displays QoS attributes when QoS is enabled in basic mode and the advanced mode is supported.

---

```
Console> show qos

Qos: basic

Basic trust: dscp
```

---

**Example 2** - The following example displays QoS attributes when QoS is enabled in basic mode on the device and the advanced mode is not supported.

```
Console> show qos  
  
Qos: disable  
  
Trust: dscp
```

## 41.4 class-map

The **class-map** command and its subcommands are used to define packet classification, marking, and aggregate policing as part of a globally-named service policy applied on a per-interface basis.

A class map consists of one or more ACLs (see [ACL Commands](#)). It defines a traffic flow by determining which packets match some or all of the criteria specified in the ACLs.

Use the **class-map** Global Configuration mode command to create or modify a class map and enter the Class-map Configuration mode (only possible when QoS is in the advanced mode).

Use the **no** form of this command to delete a class map.

All class map commands are only available when QoS is in advanced mode.

### Syntax

**class-map** *class-map-name* [*match-all* / *match-any*]

**no class-map** *class-map-name*

### Parameters

- **class-map-name**—Specifies the class map name.
- **match-all**—Performs a logical AND of all the criteria of the ACLs belonging to this class map. All match criteria in this class map must be matched.
- **match-any**—Performs a logical OR of the criteria of the ACLs belonging to this class map. Only a single match criteria in this class map must be matched.



### Default Configuration

If neither **match-all** nor **match-any** is specified, the **match-all** parameter is selected by default.

### Command Mode

Global Configuration mode

### User Guidelines

The **class-map** enters Class-map Configuration mode. In this mode, up to two **match** commands can be entered to configure the criteria for this class. Each **match** specifies an ACL.

When using two **match** commands, each must point to a different type of ACL, such as: one IP ACL and one MAC ACL. The classification is by first match, therefore, the order of the ACLs is important.

Error messages are generated in the following cases:

- There is more than one **match** command in a **match-all** class map
- There is a repetitive classification field in the participating ACLs.

After entering the Class-map Configuration mode, the following configuration commands are available:

- **exit**: Exits the Class-map Configuration mode.
- **match**: Configures classification criteria.
- **no**: Removes a match statement from a class map.

### Example

The following example creates a class map called Class1 and configures it to check that packets match all classification criteria in the ACL specified.

```
Console(config)# class-map class1 match-all  
Console(config-cmap)#match access-group acl-name
```

---

## 41.5 show class-map

The **show class-map** EXEC mode command displays all class maps when QoS is in advanced mode.

**Syntax**

**show class-map** [*class-map-name*]

**Parameters**

**class-map-name**—Specifies the name of the class map to be displayed.

**Command Mode**

EXEC mode

**Example**

The following example displays the class map for Class1.

---

```
Console> show class-map class1
Class Map match-any class1 (id4)
Match IP dscp 11 21
```

---

## 41.6 match

Use the **match** Class-map Configuration mode command to bind the Access Control Lists (ACLs) that belong to the class-map being configured. Use the **no** form of this command to delete the ACLs.

This command is available only when the device is in QoS advanced mode.

**Syntax**

**match access-group** *acl-name*

**no match access-group** *acl-name*

**Parameters**

**acl-name**—Specifies the MAC or IP ACL name.

**Default Configuration**

No match criterion is supported.

**Command Mode**

Class-map Configuration mode.

### Example

The following example defines a class map called Class1. Class1 contains an ACL called **enterprise**. Only traffic matching all criteria in **enterprise** belong to the class map.

```
Console(config)# class-map class1
Console(config-cmap)# match access-group enterprise
```

## 41.7 policy-map

A policy map contains one or more class maps and an action that is taken if the packet matches the class map. Policy maps may be bound to ports/port-channels.

Use the **policy-map** Global Configuration mode command to create a policy map and enter the Policy-map Configuration mode. Use the **no** form of this command to delete a policy map.

This command is only available when QoS is in advanced mode.

### Syntax

**policy-map** *policy-map-name*

**no policy-map** *policy-map-name*

### Parameters

**policy-map-name**—Specifies the policy map name.

### Default Configuration

N/A

### Command Mode

Global Configuration mode

### User Guidelines

Use the **policy-map** Global Configuration mode command to specify the name of the policy map to be created, added to, or modified before configuring policies for classes whose match criteria are defined in a class map.

Entering the **policy-map** Global Configuration mode command also enables configuring or modifying the class policies for that policy map. Class policies in a

policy map can be configured only if the classes have match criteria defined for them.

Policy map is applied on the ingress path.

The match criteria is for a class map. Only one policy map per interface is supported. The same policy map can be applied to multiple interfaces and directions.

The [service-policy](#) command binds a policy map to a port/port-channel.

### Example

The following example creates a policy map called Policy1 and enters the Policy-map Configuration mode.

---

```
Console(config)# policy-map policy1
Console(config-pmap)#
```

---

## 41.8 class

Use the **class** Policy-map Configuration mode command after the [policy-map](#) command to attach ACLs to a policy-map.

Use the **no** form of this command to detach a class map from a policy map.

This command is only available when QoS is in advanced mode.

### Syntax

**class** *class-map-name* [**access-group** *acl-name*]

**no class** *class-map-name*

### Parameters

- **class-map-name**—Specifies the name of an existing class map. If the class map does not exist, a new class map is created under the specified name.
- **access-group** *acl-name*—Specifies the name of an IP or MAC Access Control List (ACL).

### Default Configuration

No class map is defined for the policy map.

## Command Mode

Policy-map Configuration mode

## User Guidelines

This is the same as creating a class map and then binding it to the policy map.

You can specify an existing class map in this command, or you can use the **access-group** parameter to create a new class map.

After the policy-map is defined, use the [service-policy](#) command to attach it to a port/port-channel.

## Example

The following example defines a traffic classification (class map) called **class1** containing an ACL called **enterprise**. The class is in a policy map called **policy1**. The policy-map **policy1** now contains the ACL **enterprise**.

---

```
Console(config)# policy-map policy1
Console(config-pmap)# class class1 access-group enterprise
```

---

## 41.9 show policy-map

Use the **show policy-map EXEC** mode command to display all policy maps or a specific policy map.

This command is only available when QoS is in advanced mode.

### Syntax

```
show policy-map [policy-map-name]
```

### Parameters

**policy-map-name**—Specifies the policy map name.

### Default Configuration

All policy-maps are displayed.

### Command Mode

EXEC mode

## Example

The following example displays all policy maps.

```
Console> show policy-map

Policy Map policy1

class class1

set IP dscp 7

Policy Map policy2

class class 2

police 96000 4800 exceed-action drop

class class3

police 124000 96000 exceed-action policed-dscp-transmit
```

## 41.10 trust

Use the **trust** Policy-map Class Configuration mode command to configure the trust state. This command is relevant only when QoS is in advanced, ports-not-trusted mode. Trust indicates that traffic is sent to the queue according to the packet's QoS parameters (UP or DSCP).

Use the **no** form of this command to return to the default trust state.

This command is only available when QoS is in advanced mode.

### Syntax

**trust**

**no trust**

### Parameters

N/A

### Default Configuration

The default state is according to the mode selected in the [qos](#) command (advanced mode). The type of trust is determined in [qos advanced-mode trust](#).

## Command Mode

Policy-map Class Configuration mode

## User Guidelines

Use this command to distinguish the QoS trust behavior for certain traffic from others. For example, incoming traffic with certain DSCP values can be trusted. A class map can be configured to match and trust the DSCP values in the incoming traffic.

The type of trust is determined in [qos advanced-mode trust](#).

Trust values set with this command supersede trust values set on specific interfaces with the **qos trust** Interface Configuration mode command.

The **trust** and **set** commands are mutually exclusive within the same policy map.

Policy maps that contain **set** or **trust** commands cannot be attached, or that have Access Control List (ACL) classification to an egress interface by using the **service-policy** Interface Configuration mode command.

If specifying **trust cos**, QoS maps a packet to a queue, the received or default port CoS value, and the CoS-to-queue map.

## Example

The following example creates an ACL, places it into a class map, places the class map into a policy map and configures the trust state using the DSCP value in the ingress packet.

```
console(config)# ip access-list extended ip1
console(config-mac-a1)# permit ip any any
console(config-mac-a1)# exit
console(config)# class-map c1
console(config-cmap)# match access-group ip1
console(config-cmap)# exit
console(config)# policy-map p1
console(config-pmap)# class c1
console(config-pmap-c)# trust
```

---

## 41.11 set

Use the **set** Policy-map Class Configuration mode command to select the value that QoS uses as the DSCP, egress queue or user priority values.

This command is only available when QoS is in advanced mode.

### Syntax

**set** {*dscp new-dscp* | *queue queue-id* | *cos new-cos*}

**no set**

### Parameters

- **dscp** *new-dscp*—Specifies the new DSCP value for the classified traffic. (Range: 0–63)
- **queue** *queue-id*—Specifies the egress queue. (Range: 1-4)
- **cos** *new-cos*—Specifies the new user priority to be marked in the packet. (Range: 0–16)

### Command Mode

Policy-map Class Configuration mode

### User Guidelines

The **set** and **trust** commands are mutually exclusive within the same policy map.

To return to the Policy-map Configuration mode, use the **exit** command. To return to the Privileged EXEC mode, use the **end** command.

### Example

The following example creates an ACL, places it into a class map, places the class map into a policy map and sets the DSCP value in the packet to 56 for classes in the policy map called p1.

---

```
console(config)# ip access-list extended ip1
console(config-mac-a1)# permit ip any any
console(config-mac-a1)# exit
console(config)# class-map c1
console(config-cmap)# match access-group ip1
```



```
console(config-cmap)# exit
console(config)# policy-map p1
console(config-pmap)# class c1
Console(config-pmap-c)# set dscp 56
```

---

## 41.12 police

Use the **police** Policy-map Class Configuration mode command to define the policer for classified traffic. This defines another group of actions for the policy map (per class map).

This command is used after the **policy-map** and **class** commands.

Use the **no** form of this command to remove a policer.

This command is only available when QoS is in advanced mode.

### Syntax

**police** *committed-rate-kbps committed-burst-byte* [*exceed-action* {*drop* / *policed-dscp-transmit*}]

**no police**

### Parameters

- **committed-rate-kbps**—Specifies the average traffic rate (CIR) in kbits per second (bps). (Range: 100–10000000)
- **committed-burst-byte**—Specifies the normal burst size (CBS) in bytes. (Range: 3000–19173960)
- **exceed-action** {**drop** | **policed-dscp-transmit**}—Specifies the action taken when the rate is exceeded. The possible values are:
  - **drop**—Drops the packet.
  - **policed-dscp-transmit**—Remarks the packet DSCP, according to the policed-DSCP map as configured by the **qos map policed-dscp** Global Configuration mode command.

### Default Usage

N/A

## Command Mode

Policy-map Class Configuration mode

## User Guidelines

This command only exists in when the device is in Layer 2 mode.

Policing uses a token bucket algorithm. CIR represents the speed with which the token is added to the bucket. CBS represents the depth of the bucket.

## Example

The following example defines a policer for classified traffic. When the traffic rate exceeds 124,000 kbps and the normal burst size exceeds 9600 bytes, the packet is dropped. The class is called class1 and is in a policy map called policy1.

```
Console(config)# policy-map policy1
Console(config-pmap)# class class1
Console(config-pmap-c)# police 124000 9600 exceed-action drop
```

## 41.13 service-policy

Use the **service-policy** Interface Configuration (Ethernet, Port-channel) mode command to bind a policy map to a port/port-channel. Use the **no** form of this command to detach a policy map from an interface.

This command is only available in QoS advanced mode.

## Syntax

**service-policy** input *policy-map-name*

**no service-policy** input

## Parameters

**policy-map-name**—Specifies the policy map name to apply to the input interface. (Length: 1–32 characters)

## Command Mode

Interface Configuration (Ethernet, VLAN, Port-channel) mode

### User Guidelines

Only one policy map per interface per direction is supported.

### Example

The following example attaches a policy map called Policy1 to the input interface.

```
Console(config-if)# service-policy input policy1
```

## 41.14 qos aggregate-policer

Use the **qos aggregate-policer** Global Configuration mode command to define the policer parameters that can be applied to multiple traffic classes. Use the **no** form of this command to remove an existing aggregate policer.

This command is only available when QoS is in advanced mode.

### Syntax

**qos aggregate-policer** *aggregate-policer-name* *committed-rate-kbps* *excess-burst-byte* [*exceed-action* {*drop* | *policed-dscp-transmit*}]

**no qos aggregate-policer** *aggregate-policer-name*

### Parameters

- **aggregate-policer-name**—Specifies the aggregate policer name.
- **committed-rate-kbps**—Specifies the average traffic rate (CIR) in kbits per second (kbps). (Range: 3–57982058)
- **excess-burst-byte**—Specifies the normal burst size (CBS) in bytes. (Range: 3000–19173960)
- **exceed-action** {*drop* | *policed-dscp-transmit*}—Specifies the action taken when the rate is exceeded. The possible values are:
  - **drop**—Drops the packet.
  - **policed-dscp-transmit**—Remarks the packet DSCP.

### Default Configuration

No aggregate policer is defined.

## Command Mode

Global Configuration mode

## User Guidelines

This command only exists when the device is in Layer 2.

Define an aggregate policer if the policer aggregates traffic from multiple class maps.

Aggregate policers cannot aggregate traffic from multiple devices. If the aggregate policer is applied to more than one device, the traffic on each device is counted separately and is limited per device.

Traffic from two different ports on the same device can be aggregated for policing purposes.

An aggregate policer can be applied to multiple classes in the same policy map.

An aggregate policer cannot be deleted if it is being used in a policy map. The **no police aggregate** Policy-map Class Configuration mode command must first be used to delete the aggregate policer from all policy maps before using the **no mls qos aggregate-policer** command.

Policing uses a token bucket algorithm. CIR represents the speed with which the token is added to the bucket. CBS represents the depth of the bucket.

## Example

The following example defines the parameters of a policer called Policer1 that can be applied to multiple classes in the same policy map. When the average traffic rate exceeds 124,000 kbps or the normal burst size exceeds 9600 bytes, the packet is dropped.

```
Console(config)# qos aggregate-policer policer1 124000 9600 exceed-action drop
```

## 41.15 show qos aggregate-policer

Use the **show qos aggregate-policer EXEC** mode command to display aggregate policers

This command is only available in QoS advanced mode.

**Syntax**

**show qos aggregate-policer** [*aggregate-policer-name*]

**Parameters**

**aggregate-policer-name**—Specifies the aggregate policer name.

**Default Configuration**

All policers are displayed.

**Command Mode**

EXEC mode

**Example**

The following example displays the parameters of the aggregate policer called Policer1.

---

```
Console> show qos aggregate-policer policer1
aggregate-policer policer1 96000 4800 exceed-action drop
not used by any policy map
```

---

## 41.16 police aggregate

Use the **police aggregate** Policy-map Class Configuration mode command to apply an aggregate policer to multiple class maps within the same policy map. Use the **no** form of this command to remove an existing aggregate policer from a policy map.

This command is only available in QoS advanced mode.

**Syntax**

**police aggregate** *aggregate-policer-name*

**no police aggregate** *aggregate-policer-name*

**Parameters**

**aggregate-policer-name**—Specifies the aggregate policer name.

## Command Mode

Policy-map Class Configuration mode

## User Guidelines

An aggregate policer can be applied to multiple classes in the same policy map. An aggregate policer cannot be applied across multiple policy maps or interfaces.

Use the **exit** command to return to the Policy-map Configuration mode. Use the **end** command to return to the Privileged EXEC mode.

## Example

The following example applies the aggregate policer called Policer1 to a class called class1 in a policy map called policy1 and class2 in policy map policy2.

```
Console(config)# qos aggregate-policer policer1 124000 9600 exceed-action
drop
Console(config)# policy-map policy1
Console(config-pmap)# class class1
Console(config-pmap-c)# police aggregate policer1
Console(config-pmap-c)# exit
Console(config-pmap)# exit
Console(config)# policy-map policy2
Console(config-pmap)# class class2
Console(config-pmap-c)# police aggregate policer1
```

## 41.17 wrr-queue cos-map

Use the **wrr-queue cos-map** Global Configuration mode command to map Class of Service (CoS) values to a specific egress queue. Use the **no** form of this command to restore the default configuration.

### Syntax

```
wrr-queue cos-map queue-id cos0 ... cos7
```

```
no wrr-queue cos-map [queue-id]
```

### Parameters

- **queue-id**—Specifies the queue number to which the CoS values are mapped.
- **cos0 ... cos7**—Specifies up to 8 CoS values to map to the specified queue number. (Range: 0–7)

### Default Configuration

The default CoS value mapping to 4 queues is as follows:

CoS value 0 is mapped to queue 1.

CoS value 1 is mapped to queue 1.

CoS value 2 is mapped to queue 2.

CoS value 3 is mapped to queue 3.

CoS value 4 is mapped to queue 3.

CoS value 5 is mapped to queue 4.

CoS value 6 is mapped to queue 4.

CoS value 7 is mapped to queue 4.

### Command Mode

Global Configuration mode

### User Guidelines

Use this command to distribute traffic to different queues.

### Example

The following example maps CoS value 4 and 6 to queue 2.

---

```
Console(config)# wrr-queue cos-map 2 4 6
```

---

## 41.18 wrr-queue bandwidth

Use the **wrr-queue bandwidth** global Configuration command to assign Weighted Round Robin (WRR) weights to egress queues. The weight ratio determines the

frequency at which the packet scheduler removes packets from each queue. Use the **no** form of this command to restore the default configuration.

### Syntax

**wrr-queue bandwidth** *weight1 weight2 ... weight\_n*

**no wrr-queue bandwidth**

### Parameters

**weight1 weight1 ... weight\_n**—Specifies the ratio of bandwidth assigned by the WRR packet scheduler to the packet queues. See explanation in the User Guidelines. Separate each value by a space. (Range for each weight: 0–255)

### Default Configuration

**wrr** is disabled by default. The default wrr weight is '1' for all queues.

### Command Mode

Global Configuration mode

### User Guidelines

The ratio for each queue is defined as the queue weight divided by the sum of all queue weights (the normalized weight). This sets the bandwidth allocation of each queue.

A weight of 0 indicates that no bandwidth is allocated for the same queue, and the shared bandwidth is divided among the remaining queues. It is not recommended to set the weight of a queue to a 0 as it might stop transmission of control-protocols packets generated by the device.

All 3 queues participate in the WRR, excluding the expedite queues, whose corresponding weight is not used in the ratio calculation.

An expedite queue is a priority queue, which is serviced until empty before the other queues are serviced. The expedite queues are designated by the [priority-queue out num-of-queues](#) command.

### Example

The following assigns WRR values to the queues.

---

```
Console(config)# wrr-queue bandwidth 6 6 6 6
```



---

## 41.19 priority-queue out num-of-queues

An expedite queue is a strict priority queue, which is serviced until empty before the other lower priority queues are serviced.

Use the **priority-queue out num-of-queues** Global Configuration mode command to configure the number of expedite queues. Use the **no** form of this command to restore the default configuration.

### Syntax

**priority-queue out num-of-queues** *number-of-queues*

**no priority-queue out num-of-queues**

### Parameters

**number-of-queues**—Specifies the number of expedite (strict priority) queues. Expedite queues are assigned to the queues with the higher indexes. (Range: 0–4). There must be either 0 wrr queues or more than one.

If **number-of-queues** = 0, all queues are assured forwarding (according to wrr weights) If the **number-of-queues** = 4, all queues are expedited (strict priority queues).

### Default Configuration

All queues are expedite queues.

### Command Mode

Global Configuration mode

### User Guidelines

the weighted round robin (WRR) weight ratios are affected by the number of expedited queues, because there are fewer queues participating in WRR. This indicates that the corresponding weight in the **wrr-queue bandwidth** Interface Configuration mode command is ignored (not used in the ratio calculation).

### Example

The following example configures the number of expedite queues as 2.

---

```
Console(config)# priority-queue out num-of-queues 2
```

---

## 41.20 traffic-shape

The egress port shaper controls the traffic transmit rate (Tx rate) on a port.

Use the **traffic-shape** Interface Configuration mode command to configure the egress port shaper. Use the **no** form of this command to disable the shaper.

### Syntax

**traffic-shape** *committed-rate* [*committed-burst*]

**no traffic-shape**

### Parameters

- **committed-rate**—Specifies the maximum average traffic rate (CIR) in kbits per second (kbps). (Range: GE: 64kbps–maximum port speed; 10GE: 64Kbps–maximum port speed)
- **committed-burst**—Specifies the maximum permitted excess burst size (CBS) in bytes. (Range: 4096 - 16762902 bytes)

### Default Configuration

The shaper is disabled.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### Example

The following example sets a traffic shaper on `gi5` on queue 1 when the average traffic rate exceeds 124000 kbps or the normal burst size exceeds 9600 bytes.

```
Console(config)# interface gi5
Console(config-if)# traffic-shape 1 124000 9600
```

---

## 41.21 traffic-shape queue

The egress port shaper controls the traffic transmit rate (Tx rate) on a queue on a port.

Use the **traffic-shape queue** Interface Configuration mode command to configure the egress queue shaper. Use the **no** form of this command to disable the shaper.

### Syntax

**traffic-shape queue** *queue-id committed-rate* [*committed-burst*]

**no traffic-shape queue** *queue-id*

### Parameters

- **queue-id**—Specifies the queue number to which the shaper is assigned. (Range: 1-4)
- **committed-rate**—Specifies the average traffic rate (CIR) in kbits per second (kbps). (Range: 64 kbps–maximum port speed)
- **committed-burst**—Specifies the excess burst size (CBS) in bytes. (Range: 4096 - 16762902 bytes)

### Default Configuration

The shaper is disabled.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### Example

The following example sets a shaper on queue 1 on `gi5` when the average traffic rate exceeds 124000 kbps or the normal burst size exceeds 9600 bytes.

```
Console(config)# interface gi5
Console(config-if)# traffic-shape 1 124000 9600
```

---

## 41.22 rate-limit (Ethernet)

Use the **rate-limit** Interface Configuration mode command to limit the incoming traffic rate on a port. Use the **no** form of this command to disable the rate limit.

### Syntax

**rate-limit** *committed-rate-kbps* [*burst committed-burst-bytes*]

**no rate-limit**

### Parameters

- **committed-rate-kbps**—Specifies the maximum number of kilobits per second of ingress traffic on a port. The range is 100 – max port speed.
- **burst committed-burst-bytes**—The burst size in bytes (3000–19 173 960). If unspecified, defaults to 128K.

### Default Configuration

Rate limiting is disabled.

Committed-burst-bytes is 128K.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

Storm control and rate-limit (of Unicast packets) cannot be enabled simultaneously on the same port.

### Example

The following example limits the incoming traffic rate on `gi5` to 150,000 kbps.

```
Console(config)# interface gi5
Console(config-if)# rate-limit 150000
```

## 41.23 rate-limit (VLAN)

Use the Layer 2 **rate-limit (VLAN)** Global Configuration mode command to limit the incoming traffic rate for a VLAN. Use the **no** form of this command to disable the rate limit.

### Syntax

**rate-limit** *vlan-id committed-rate committed-burst*

**no rate-limit** *vlan*

### Parameters

- **vlan-id**—Specifies the VLAN ID.

- **committed-rate**—Specifies the average traffic rate (CIR) in kbits per second (kbps). (Range: 3-57982058)
- **committed-burst**—Specifies the maximum burst size (CBS) in bytes. (Range: 3000-19173960)

### Default Configuration

Rate limiting is disabled.

Committed-burst-bytes is 128K.

### Command Mode

Global Configuration mode

### User Guidelines

Traffic policing in a policy map takes precedence over VLAN rate limiting. If a packet is subject to traffic policing in a policy map and is associated with a VLAN that is rate limited, the packet is counted only in the traffic policing of the policy map.

This command does not work in Layer 3 mode.

### Example

The following example limits the rate on VLAN 11 to 150000 kbps or the normal burst size to 9600 bytes.

---

```
Console(config)# rate-limit 11 150000 9600
```

---

## 41.24 qos wrr-queue wrtd

Use the **qos wrr-queue wrtd** Global Configuration mode command to enable Weighted Random Tail Drop (WRTD). Use the **no** form of this command to disable WRTD.

### Syntax

**qos wrr-queue wrtd**

**no qos wrr-queue wrtd**

**Parameters**

N/A

**Default**

Disabled

**Command Mode**

Global Configuration mode

**User Guidelines**

The command is effective after reset.

**Example**

```
Console(conf)#>qos wrr-queue wrtd
```

This setting will take effect only after copying running configuration to startup configuration and resetting the device

```
Console(config)#
```

---

**41.25 show qos wrr-queue wrtd**

Use the **show qos wrr-queue wrtd** Exec mode command to display the Weighted Random Tail Drop (WRTD) configuration.

**Syntax**

```
show qos wrr-queue wrtd
```

**Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

Exec mode

---

## Example

```
Console> show qos wrr-queue wrtd
```

```
Weighted Random Tail Drop is disabled
```

```
Weighted Random Tail Drop will be enabled after reset
```

---

## 41.26 show qos interface

Use the **show qos interface** EXEC mode command to display Quality of Service (QoS) information on the interface.

### Syntax

```
show qos interface [buffers / queueing / policers / shapers / rate-limit] [interface-id]
```

### Parameters

- **buffers**—Displays the buffer settings for the interface's queues. For GE ports, displays the queue depth for each of the 4 queues.
- **queueing**—Displays the queue's strategy (WRR or EF), the weight for WRR queues, the CoS to queue map and the EF priority.
- **policers**—Displays all the policers configured for this interface, their settings, and the number of policers currently unused (on a VLAN).
- **shapers**—Displays the shaper of the specified interface and the shaper for the queue on the specified interface.
- **rate-limit**—Displays the rate-limit configuration.
- **interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port, or Port-channel.

### Default Configuration

N/A

### Command Mode

EXEC mode

## User Guidelines

If no parameter is specified with the **show qos interface** command, the port QoS mode (DSCP trusted, CoS trusted, untrusted, and so on), default CoS value, DSCP-to-DSCP- map (if any) attached to the port, and policy map (if any) attached to the interface are displayed. If a specific interface is not specified, the information for all interfaces is displayed.

## Example

This is an example of the output from the **show qos interface queueing** command for 4 queues.

---

```
Console> show qos interface queueing gil
```

```
gil
```

```
wrr bandwidth weights and EF priority:
```

```
qid  weights  Ef          Priority
```

```
1    125      Disable    N/A
```

```
2    125      Disable    N/A
```

```
3    125      Disable    N/A
```

```
4    125      Disable    N/A
```

```
Cos-queue map:
```

```
CoS  QID
```

```
0    2
```

```
1    1
```

```
2    1
```

```
3    2
```

```
4    3
```

```
5    3
```

```
6    4
```

```
7    4
```



This is an example of the output from the **show qos interface policer** command.

---

```
Console> show qos interface policer gi1

Ethernet gi1

Class map: A

Policer type: aggregate

Committed rate: 192000 bps

Committed burst: 9600 bytes

Exceed-action: policed-dscp-transmit

Class map: B

Policer type: single

Committed rate: 192000 bps

Committed burst: 9600 bytes

Exceed-action: drop

Class map: C

Policer type: none

Committed rate: N/A

Committed burst: N/A

Exceed-action: N/A
```

This is an example of the output from the **show qos interface rate-limit** command.

---

```
Console> show qos interface rate-limit gi1

Port          rate-limit [kbps]      Burst [KBytes]
-----
gi1           1000                   512K
```

---

## 41.27 wrr-queue

Use the **wrr-queue** Global Configuration mode command to enable the tail-drop mechanism on an egress queue. Use the **no** form of this command to disable the tail-drop mechanism on an egress queue.

**Syntax**

**wrr-queue** *tail-drop*

**no wrr-queue**

**Parameters**

**tail-drop**— Specifies the tail-drop mechanism.

**Default Configuration**

The tail-drop mechanism on an egress queue is disabled.

**Command Mode**

Global Configuration mode

**User Guidelines**

This command can only be used if Advanced mode is enabled.

**Example**

The following example enables the tail-drop mechanism on an egress queue.

---

```
Console(config)# wrr-queue tail-drop
```

---

**41.28 qos wrr-queue threshold**

Use the **qos wrr-queue threshold** Global Configuration mode command to assign queue thresholds globally. Use the **no** form of this command to restore the default configuration.

This command is only available in QoS advanced mode.

**Syntax**

**qos wrr-queue threshold** gigabitethernet *queue-id* *threshold-percentage*

**no qos wrr-queue threshold** gigabitethernet *queue-id*

### Parameters

- **gigabitethernet**—Specifies that the thresholds are to be applied to Gigabit Ethernet ports.
- **queue-id**—Specifies the queue number to which the tail-drop threshold is assigned.
- **threshold-percentage**—Specifies the queue threshold percentage value.

### Default Configuration

The default threshold is 80 percent.

### Command Mode

Global Configuration mode

### User Guidelines

If the threshold is exceeded, packets with the corresponding Drop Precedence (DP) are dropped until the threshold is no longer exceeded.

### Example

The following example assigns a threshold of 80 percent to WRR queue 1.

---

```
Console(config)# qos wrr-queue threshold gigabitethernet 1 80
```

---

## 41.29 qos map policed-dscp

Use the **qos map policed-dscp** Global Configuration mode command to configure the policed-DSCP map for remarking purposes. Use the **no** form of this command to restore the default configuration.

This command is only available in QoS advanced mode.

### Syntax

**qos map policed-dscp** *dscp-list* **to** *dscp-mark-down*

**no qos map policed-dscp** [*dscp-list*]

### Parameters

- **dscp-list**—Specifies up to 8 DSCP values, separated by spaces. (Range: 0–63)
- **dscp-mark-down**—Specifies the DSCP value to mark down. (Range: 0–63)

### Default Configuration

The default map is the Null map, which means that each incoming DSCP value is mapped to the same DSCP value.

### Command Mode

Global Configuration mode.

### User Guidelines

The original DSCP value and policed-DSCP value must be mapped to the same queue in order to prevent reordering.

### Example

The following example marks incoming DSCP value 3 as DSCP value 5 on the policed-DSCP map.

---

```
Console(config)# qos map policed-dscp 3 to 5
```

---

## 41.30 qos map dscp-queue

Use the **qos map dscp-queue** Global Configuration mode command to configure the DSCP to CoS map. Use the **no** form of this command to restore the default configuration.

### Syntax

```
qos map dscp-queue dscp-list to queue-id
```

```
no qos map dscp-queue [dscp-list]
```

### Parameters

- **dscp-list**—Specifies up to 8 DSCP values, separated by spaces. (Range: 0–63)

- **queue-id**—Specifies the queue number to which the DSCP values are mapped.

### Default Configuration

The default map for 4 queues is as follows.

### Command Mode

Global Configuration mode

### Example

The following example maps DSCP values 33, 40 and 41 to queue 1.

---

```
Console(config)# qos map dscp-queue 33 40 41 to 1
```

---

## 41.31 qos map dscp-dp

Use the **qos map dscp-dp** Global Configuration mode command to map the DSCP values to Drop Precedence. Use the **no** form of this command to restore the default configuration.

This command is only available in QoS advanced mode.

### Syntax

```
qos map dscp-dp dscp-list to dp
```

```
no qos map dscp-dp [dscp-list]
```

### Parameters

- **dscp-list**—Specifies up to 8 DSCP values, with values separated by a space. (Range: 0–63)
- **dp**—Specifies the Drop Precedence value to which the DSCP values are mapped. (values: 0,2) where 2 is the highest Drop Precedence).

### Default Configuration

All the DSCPs are mapped to Drop Precedence 0.

### Command Mode

Global Configuration mode.

### Example

The following example maps DSCP values 25, 27 and 29 to Drop Precedence 2.

```
Console(config)# qos map dscp-dp 25 27 29 to 2
```

## 41.32 qos trust (Global)

Use the **qos trust** Global Configuration mode command to configure the system to the basic mode and trust state. Use the **no** form of this command to return to the default configuration.

### Syntax

```
qos trust {cos / dscp}
```

```
no qos trust
```

### Parameters

- **cos**— Specifies that ingress packets are classified with packet CoS values. Untagged packets are classified with the default port CoS value.
- **dscp**— Specifies that ingress packets are classified with packet DSCP values.

### Default Configuration

DSCP is the default trust mode.

### Command Mode

Global Configuration mode

### User Guidelines

This command can be used only in QoS basic mode.

Packets entering a Quality of Service (QoS) domain are classified at the edge of the QoS domain. When the packets are classified at the edge, the switch port within the QoS domain can be configured to one of the trusted states because there is no need to classify the packets at every switch within the domain.

Use this command to specify whether the port is trusted and which fields of the packet to use to classify traffic.

When the system is configured with trust DSCP, the traffic is mapped to the queue by the DSCP-queue map.

When the system is configured with trust CoS, the traffic is mapped to the queue by the CoS-queue map.

For an inter-QoS domain boundary, configure the port to the DSCP-trusted state and apply the DSCP-to-DSCP-mutation map if the DSCP values are different in the QoS domains.

### Example

The following example configures the system to the DSCP trust state.

---

```
Console(config)# qos trust dscp
```

---

## 41.33 qos trust (Interface)

Use the **qos trust** Interface Configuration (Ethernet, Port-channel) mode command to enable port trust state while the system is in the basic QoS mode. Use the **no** form of this command to disable the trust state on each port.

### Syntax

**qos trust**

**no qos trust**

### Default Configuration

Each port is enabled while the system is in basic mode.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### Example

The following example configures `gi 15` to the default trust state.

---

```
Console(config)# interface gi5
```

```
Console(config-if)# qos trust
```

---

### 41.34 qos cos

Use the **qos cos** Interface Configuration (Ethernet, Port-channel) mode command to define the default CoS value of a port. Use the **no** form of this command to restore the default configuration.

#### Syntax

**qos cos** *default-cos*

**no qos cos**

#### Parameters

**default-cos**—Specifies the default CoS value (VPT value) of the port. If the port is trusted and the packet is untagged, then the default CoS value become the CoS value. (Range: 0–16)

#### Default Configuration

The default CoS value of a port is 0.

#### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

#### User Guidelines

Use the default CoS value to assign a CoS value to all untagged packets entering the interface.

#### Example

The following example defines the port `gi5` default CoS value as 3 .

---

```
Console(config)# interface gi5
```

```
Console(config-if)# qos cos 3
```

---

### 41.35 qos dscp-mutation

Use the **qos dscp-mutation** Global Configuration mode command to apply the DSCP Mutation map to system DSCP trusted ports. Use the **no** form of this command to restore the trusted port with no DSCP mutation.



**Syntax****qos dscp-mutation****no qos dscp-mutation****Command Mode**

Global Configuration mode.

**User Guidelines**

Apply the DSCP-to-DSCP-mutation map to a port at the boundary of a Quality of Service (QoS) administrative domain. If two QoS domains have different DSCP definitions, use the DSCP-to-DSCP-mutation map to translate a set of DSCP values to match the definition of another domain. Apply the map to ingress and to DSCP-trusted ports only. Applying this map to a port causes IP packets to be rewritten with newly mapped DSCP values at the ingress ports. If applying the DSCP mutation map to an untrusted port, to class of service (CoS), or to an IP-precedence trusted port.

Global trust mode must be DSCP or CoS-DSCP. In advanced CoS mode, ports must be trusted.

**Example**

The following example applies the DSCP Mutation map to system DSCP trusted ports.

---

```
Console(config)# qos dscp-mutation
```

---

**41.36 qos map dscp-mutation**

Use the **qos map dscp-mutation** Global Configuration mode command to configure the DSCP to DSCP Mutation table. Use the **no** form of this command to restore the default configuration.

**Syntax****qos map dscp-mutation** *in-dscp* to *out-dscp***no qos map dscp-mutation** [*in-dscp*]

### Parameters

- **in-dscp**—Specifies up to 8 DSCP values to map, separated by spaces. (Range: 0–63)
- **out-dscp**—Specifies up to 8 DSCP mapped values, separated by spaces. (Range: 0–63)

### Default Configuration

The default map is the Null map, which means that each incoming DSCP value is mapped to the same DSCP value.

### Command Mode

Global Configuration mode.

### User Guidelines

This is the only map that is not globally configured. It is possible to have several maps and assign each one to a different port.

### Example

The following example changes DSCP values 1, 2, 4, 5 and 6 to DSCP Mutation Map value 63.

---

```
Console(config)# qos map dscp-mutation 1 2 4 5 6 to 63
```

---

## 41.37 show qos map

Use the **show qos map** EXEC mode command to display the various types of QoS mapping.

### Syntax

```
show qos map [dscp-queue | dscp-dp | policed-dscp | dscp-mutation]
```

### Parameters

- **dscp-queue**—Displays the DSCP to queue map.
- **dscp-dp**—Displays the DSCP to Drop Precedence map.
- **policed-dscp**—Displays the DSCP to DSCP remark table.

- **dscp-mutation**—Displays the DSCP-DSCP mutation table.

### Command Mode

EXEC mode

### Example

The following example displays the QoS mapping information.

---

```

Console> show qos map dscp-queue

Dscp-queue map:
d1  :   d2  0   1   2   3   4   5   6   7   8   9
--  --  --  --  --  --  --  --  --  --  --  --
0   :           01  01  01  01  01  01  01  01  01  01  01
1   :           01  01  01  01  01  01  01  01  02  02  02
2   :           02  02  02  02  03  03  03  03  03  03
3   :           04  04  05  05  05  05  05  05  05  05
4   :           06  06  06  06  06  06  06  06  07  07
5   :           07  07  07  07  07  07  08  08  08  08
6   :           08  08  08  08

```

---

## 41.38 clear qos statistics

Use the **clear qos statistics** EXEC mode command to clear the QoS statistics counters.

### Syntax

**clear qos statistics**

### Command Mode

EXEC mode

### Example

The following example clears the QoS statistics counters.

---

```

Console# clear qos statistics

```

---

### 41.39 qos statistics policer

Use the **qos statistics policer** Interface Configuration (Ethernet, Port-channel) mode command to enable counting in-profile and out-of-profile. Use the **no** form of this command to disable counting.

#### Syntax

**qos statistics policer** *policy-map-name class-map-name*

**no qos statistics policer** *policy-map-name class-map-name*

#### Parameters

- **policy-map-name**—Specifies the policy map name.
- **class-map-name**—Specifies the class map name.

#### Default Configuration

Counting in-profile and out-of-profile is disabled.

#### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

#### Example

The following example enables counting in-profile and out-of-profile on the interface.

---

```
Console(config-if)# qos statistics policer policy1 class1
```

---

### 41.40 qos statistics aggregate-policer

Use the **qos statistics aggregate-policer** Global Configuration mode command to enable counting in-profile and out-of-profile. Use the **no** form of this command to disable counting.

#### Syntax

**qos statistics aggregate-policer** *aggregate-policer-name*

**no qos statistics aggregate-policer** *aggregate-policer-name*

### Parameters

**aggregate-policer-name**—Specifies the aggregate policer name.

### Default Configuration

Counting in-profile and out-of-profile is disabled.

### Command Mode

Global Configuration mode

### Example

The following example enables counting in-profile and out-of-profile on the interface.

---

```
Console(config)# qos statistics aggregate-policer policer1
```

---

## 41.41 qos statistics queues

Use the **qos statistics queues** Global Configuration mode command to enable QoS statistics for output queues. Use the **no** form of this command to disable QoS statistics for output queues.

### Syntax

```
qos statistics queues set {queue | all} {dp | all} {interface | all}
```

```
no qos statistics queues set
```

### Parameters

- **set**—Specifies the counter set number.
- **interface**—Specifies the Ethernet port.
- **queue**—Specifies the output queue number.
- **dp**—Specifies the drop precedence. The available values are: **high**, **low**.

### Default Configuration

Set 1: All interfaces, all queues, high DP.

Set 2: All interfaces, all queues, low DP.

**Command Mode**

Global Configuration mode

**User Guidelines**

There are no user guidelines for this command.

If the queue parameter is all, traffic in cascading ports is also counted.

**Example**

The following example enables QoS statistics for output queues for counter set 1.

---

```
Console(config)# qos statistics queues 1 all all all
```

---

**41.42 show qos statistics**

Use the **show qos statistics** EXEC mode command to display Quality of Service statistical information.

**Syntax**

**show qos statistics**

**Command Mode**

EXEC mode

**User Guidelines**

Up to 16 sets of counters can be enabled for policers. The counters can be enabled in the creation of the policers.

Use the **qos statistics queues** Global Configuration mode command to enable QoS statistics for output queues.

**Example**

The following example displays Quality of Service statistical information.

```

Console# show qos statistics

Policers
-----

Interface  Policy  Class  In-profile  Out-of-prof
          map   Map    bytes      ile bytes
-----
gi1        -       Class1 7564575    52
gi1        Policy1 Class2 8759       3214
gi2        Policy1 Class1 746587458 23
gi2        Policy1 Class2 5326
          Policy1

Aggregate Policers
-----

Name          In-profile bytes  Out-of-profile bytes
-----
Policer1     7985687           121322

Output Queues
-----

Interface  Queue  DP  Total  %TD packets
          -----
gi1        2      High  -----  1.2%
gi2        All    High  799921  0.2%
          5387326

```

### 41.43 security-suite enable

Use the **security-suite enable** Global Configuration mode command to enable the security suite feature. This feature supports protection against various types of attacks.

When this command is used, hardware resources are reserved. These hardware resources are released when the **no security-suite enable** command is entered.

The security-suite feature can be enabled in one of the following ways:

- **Global-rules-only** - This enables the feature globally but per-interface features are not enabled.
- **All** (no keyword) - The feature is enabled globally and per-interface.

Use the **no** form of this command to disable the security suite feature.

When security-suite is enabled, you can specify the types of protection required. The following commands can be used:

- `security-suite dos protect`
- `security-suite dos syn-attack`
- `security-suite deny martian-addresses`
- `security-suite deny syn`
- `security-suite deny icmp`
- `security-suite deny fragmented`
- `show security-suite configuration`
- `security-suite dos protect`

#### Syntax

**security-suite enable** [*global-rules-only*]

**no security-suite enable**

#### Parameters

**global-rules-only**—Specifies that all the security suite commands are global commands only (they cannot be applied per-interface). This setting saves space in the Ternary Content Addressable Memory (TCAM). If this keyword is not used, security-suite commands can be used both globally on per-interface.



### Default Configuration

The security suite feature is disabled.

If **global-rules-only** is not specified, the default is to enable security-suite globally and per interfaces.

### Command Mode

Global Configuration mode

### User Guidelines

MAC ACLs must be removed before the security-suite is enabled. The rules can be re-entered after the security-suite is enabled.

If ACLs or policy maps are assigned on interfaces, per interface security-suite rules cannot be enabled.

### Examples

**Example 1** - The following example enables the security suite feature and specifies that security suite commands are global commands only. When an attempt is made to configure security-suite on a port, it fails.

---

```
Console(config)# security-suite enable global-rules-only
```

```
Console(config)# interface gil
```

```
Console(config-if)# security-suite dos syn-attack 199 any /10
```

To perform this command, DoS Prevention must be enabled in the per-interface mode.

---

**Example 2** - The following example enables the security suite feature globally and on interfaces. The security-suite command succeeds on the port.

```
Console(config)# security-suite enable
```

```
Console(config)# interface gil
```

```
Console(config-if)# security-suite dos syn-attack 199 any /10
```

```
Console(config-if)#
```

---

## 41.44 security-suite dos protect

Use the **security-suite dos protect** Global Configuration mode command to protect the system from specific well-known Denial of Service (DoS) attacks.

There are three types of attacks against which protection can be supplied (see parameters below).

Use the **no** form of this command to disable DoS protection.

### Syntax

**security-suite dos protect** *{add attack / remove attack}*

**no security-suite dos protect**

### Parameters

- **add/remove attack**—Specifies the attack type to add/remove. To add an attack is to provide protection against it; to remove the attack is to remove protection.

The possible attack types are:

- **stacheldraht**—Discards TCP packets with source TCP port 16660.
- **invasor-trojan**—Discards TCP packets with destination TCP port 2140 and source TCP port 1024.
- **back-orifice-trojan**—Discards UDP packets with destination UDP port 31337 and source UDP port 1024.

### Default Configuration

No protection is configured.

### Command Mode

Global Configuration mode

### User Guidelines

For this command to work, [security-suite enable](#) must be enabled globally.

### Example

The following example protects the system from the Invasor Trojan DOS attack.

---

```
Console(config)# security-suite dos protect add invasor-trojan
```

## 41.45 security-suite dos syn-attack

Use the **security-suite dos syn-attack** Interface Configuration mode command to rate limit Denial of Service (DoS) SYN attacks. This provides partial blocking of SYN packets (up to the rate that the user specifies).

Use the **no** form of this command to disable rate limiting.

### Syntax

```
security-suite dos syn-attack syn-rate {any | ip-address} {mask | prefix-length}
```

```
no security-suite dos syn-attack {any | ip-address} {mask | prefix-length}
```

### Parameters

- **syn-rate**—Specifies the maximum number of connections per second. (Range: 199–1000)
- **any | ip-address**—Specifies the destination IP address. Use **any** to specify all IP addresses.
- **mask**—Specifies the network mask of the destination IP address.
- **prefix-length**—Specifies the number of bits that comprise the destination IP address prefix. The prefix length must be preceded by a forward slash (/).

### Default Configuration

No rate limit is configured.

If **ip-address** is unspecified, the default is 255.255.255.255

If **prefix-length** is unspecified, the default is 32.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

For this command to work, [security-suite enable](#) must be enabled both globally and for interfaces.

This command rate limits ingress TCP packets with "SYN=1", "ACK=0" and "FIN=0" for the specified destination IP addresses.

SYN attack rate limiting is implemented after the security suite rules are applied to the packets. The ACL and QoS rules are not applied to those packets.

Since the hardware rate limiting counts bytes, it is assumed that the size of “SYN” packets is short.

### Example

The following example attempts to rate limit DoS SYN attacks on a port. It fails because security suite is enabled globally and not per interface.

---

```
Console(config)# security-suite enable global-rules-only
```

```
Console(config)# interface gil
```

```
Console(config-if)# security-suite dos syn-attack 199 any /10
```

To perform this command, DoS Prevention must be enabled in the per-interface mode.

---

## 41.46 security-suite deny martian-addresses

Use the **security-suite deny martian-addresses** Global Configuration mode command to deny packets containing system-reserved IP addresses or user-defined IP addresses.

### Syntax

**security-suite deny martian-addresses** *{add {ip-address {mask /prefix-length}} | remove {ip-address {mask /prefix-length}}* (Add/remove user-specified IP addresses)

**security-suite deny martian-addresses reserved** *{add | remove}* (Add/remove system-reserved IP addresses, see tables below)

**no security-suite deny martian-addresses** ( This command removes addresses reserved by **security-suite deny martian-addresses** *{add {ip-address {mask /prefix-length}} | remove {ip-address {mask /prefix-length}}*, and removes all entries added by the user.

The user can remove a specific entry by using **remove ip-address {mask /prefix-length}** parameter.

There is no **no** form of the **security-suite deny martian-addresses reserved** *{add | remove}* command. Use instead the **security-suite deny martian-addresses reserved remove** command to remove protection (and free up hardware resources).

### Parameters

- **reserved add/remove**—Add or remove the table of reserved addresses below.
- **ip-address**—Adds/discards packets with the specified IP source or destination address.
- **mask**—Specifies the network mask of the IP address.
- **prefix-length**—Specifies the number of bits that comprise the IP address prefix. The prefix length must be preceded by a forward slash (/).
- **reserved**—Discards packets with the source or destination IP address in the block of the reserved (Martian) IP addresses. See the User Guidelines for a list of reserved addresses.

### Default Configuration

Martian addresses are allowed.

### Command Mode

Global Configuration mode

### User Guidelines

For this command to work, [security-suite enable](#) must be enabled globally.

**security-suite deny martian-addresses *reserved*** adds or removes the addresses in the following table:

Address block	Present use
<b>0.0.0.0/8 (except when 0.0.0.0/32 is the source address)</b>	Addresses in this block refer to source hosts on "this" network.
<b>127.0.0.0/8</b>	This block is assigned for use as the Internet host loopback address.
<b>192.0.2.0/24</b>	This block is assigned as "TEST-NET" for use in documentation and example code.

Address block	Present use
<b>224.0.0.0/4 as source</b>	This block, formerly known as the Class D address space, is allocated for use in IPv4 multicast address assignments.
<b>240.0.0.0/4 (except when 255.255.255.255/3 2 is the destination address)</b>	This block, formerly known as the Class E address space, is reserved.

Note that if the reserved addresses are included, individual reserved addresses cannot be removed.

### Example

The following example discards all packets with a source or destination address in the block of the reserved IP addresses.

```
Console(config)# security-suite deny martian-addresses reserved add
```

## 41.47 security-suite deny syn

Use the **security-suite deny syn** Interface Configuration (Ethernet, Port-channel) mode command to block the creation of TCP connections from a specific interface. This a complete block of these connections.

Use the **no** form of this command to permit creation of TCP connections.

### Syntax

```
security-suite deny syn { [add {tcp-port | any} {ip-address | any} {mask /  
/prefix-length} ] |  
[remove {tcp-port | any} {ip-address | any} {mask /prefix-length}] }
```

```
no security-suite deny syn
```

### Parameters

- **ip-address | any**—Specifies the destination IP address. Use **any** to specify all IP addresses.
- **mask**— Specifies the network mask of the destination IP address.
- **prefix-length**—Specifies the number of bits that comprise the destination IP address prefix. The prefix length must be preceded by a forward slash (/).

- **tcp-port | any**—Specifies the destination TCP port. The possible values are: **http**, **ftp-control**, **ftp-data**, **ssh**, **telnet**, **smtp**, **dns**, **tftp**, **ntp**, **snmp** or **port number**. Use **any** to specify all ports.

### Default Configuration

Creation of TCP connections is allowed from all interfaces.

If the **mask** is not specified, it defaults to 255.255.255.255.

If the **prefix-length** is not specified, it defaults to 32.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

For this command to work, **security-suite enable** must be enabled both globally and for interfaces.

The blocking of TCP connection creation from an interface is done by discarding ingress TCP packets with "SYN=1", "ACK=0" and "FIN=0" for the specified destination IP addresses and destination TCP ports.

### Example

The following example attempts to block the creation of TCP connections from an interface. It fails because security suite is enabled globally and not per interface.

---

```
Console(config)# security-suite enable global-rules-only
```

```
Console(config)# interface gil
```

```
Console(config-if)# security-suite deny syn add any /32 any
```

To perform this command, DoS Prevention must be enabled in the per-interface mode.

---

## 41.48 security-suite deny icmp

Use the **security-suite deny icmp** Interface Configuration (Ethernet, Port-channel) mode command to discard ICMP echo requests from a specific interface (to prevent attackers from knowing that the device is on the network).

Use the **no** form of this command to permit echo requests.

## Syntax

**security-suite deny icmp** *{{add {ip-address | any} {mask /prefix-length}} | [remove {ip-address | any} {mask /prefix-length}]}*

**no security-suite deny icmp**

## Parameters

- **ip-address | any**—Specifies the destination IP address. Use **any** to specify all IP addresses.
- **mask**—Specifies the network mask of the IP address.
- **prefix-length**—Specifies the number of bits that comprise the IP address prefix. The prefix length must be preceded by a forward slash (/).

## Default Configuration

Echo requests are allowed from all interfaces.

If **mask** is not specified, it defaults to 255.255.255.255.

If **prefix-length** is not specified, it defaults to 32.

## Command Mode

Interface Configuration (Ethernet, Port-channel) mode

## User Guidelines

For this command to work, [security-suite enable](#) must be enabled both globally and for interfaces.

This command discards ICMP packets with "ICMP type= Echo request" that ingress the specified interface.

## Example

The following example attempts to discard echo requests from an interface.

```
Console(config)# security-suite enable global-rules-only
Console(config)# interface g1
Console(config-if)# security-suite deny icmp add any /32
```

To perform this command, DoS Prevention must be enabled in the per-interface mode.



## 41.49 security-suite deny fragmented

Use the **security-suite deny fragmented** Interface Configuration (Ethernet, Port-channel) mode command to discard IP fragmented packets from a specific interface.

Use the **no** form of this command to permit IP fragmented packets.

### Syntax

```
security-suite deny fragmented [[add {ip-address | any} {mask | /prefix-length}] |  
[remove {ip-address | any} {mask | /prefix-length}]
```

```
no security-suite deny fragmented
```

### Parameters

- **ip-address | any**—Specifies the destination IP address. Use **any** to specify all IP addresses.
- **mask**—Specifies the network mask of the IP address.
- **prefix-length**—Specifies the number of bits that comprise the IP address prefix. The prefix length must be preceded by a forward slash (/).

### Default Configuration

Fragmented packets are allowed from all interfaces.

If **mask** is unspecified, the default is 255.255.255.255.

If **prefix-length** is unspecified, the default is 32.

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

For this command to work, [security-suite enable](#) must be enabled both globally and for interfaces.

### Example

The following example attempts to discard IP fragmented packets from an interface.

```

Console(config)# security-suite enable global-rules-only
Console(config)# interface gil
Console(config-if)# security-suite deny fragmented add any /32

```

To perform this command, DoS Prevention must be enabled in the per-interface mode.

---

## 41.50 show security-suite configuration

Use the **show security-suite configuration** EXEC mode command to display the security-suite configuration.

### Syntax

**show security-suite configuration**

### Command Mode

EXEC mode

### Example

The following example displays the security-suite configuration.

---

```

Console# show security-suite configuration

```

```

Security suite is enabled (Per interface rules are
enabled).

```

```

Denial Of Service Protect: stacheldraht, invasor-trojan,
back-office-trojan.

```

```

Denial Of Service SYN Attack
Interface          IP Address          SYN Rate(pps)
-----
gil                176.16.23.0\24     100

```

## Martian addresses filtering

Reserved addresses: enabled.

Configured addresses: 10.0.0.0/8, 192.168.0.0/16

## SYN filtering

Interface	IP Address	TCP port
-----	-----	-----
gi2	176.16.23.0\24	FTP

## ICMP filtering

Interface	IP Address
-----	-----
gi2	176.16.23.0\24

## Fragmented packets filtering

Interface	IP Address
-----	-----
gi2s	176.16.23.0\24

## 42 Voice VLAN Commands

### 42.1 voice vlan state

The **voice vlan state** Global Configuration mode command sets the type of voice VLAN that is functional on the device or disables voice VLAN entirely. The **no** format of the command returns to the default.

#### Syntax

```
voice vlan state {oui-enabled | auto-enabled | auto-triggered | disabled}
```

```
no voice vlan state
```

#### Parameters

- **oui-enabled**—Voice VLAN is of type OUI on the device.
- **auto-enabled**—Auto VLAN is enabled.
- **auto-triggered**—Voice VLAN is enabled, but it will be triggered by an external event, such as an IP phone being attached to a Smartport.
- **disabled**—Voice VLAN is disabled.

#### Default Configuration

```
auto-triggered
```

#### Command Mode

```
Global Configuration mode
```

#### User Guidelines

By factory default, CDP, LLDP, and LLDP-MED are enabled on the switch. In addition, manual Smartport mode and Basic QoS with trusted DSCP is enabled.

All ports are members of default VLAN 1, which is also the default Voice VLAN.

In addition, dynamic voice VLAN (**auto-triggered**) mode is the default mode of auto voice VLAN. In this mode, voice VLAN is enabled by a trigger (advertisement received by voice device attached to port).

If the administrative state is:

- **disabled** — The operational state is **disabled**.

- **oui-enabled** —The operational state is **oui-enabled**.
- **auto-enabled** — The operational state is **auto-enabled**.
- **auto-triggered** — The operational state is **auto-triggered** only if:
  - A static local configured voice VLAN ID, CoS/802.1p, and/or DSCP that is not factory default is configured.
  - A CDP voice VLAN advertisement is received from a neighboring switch. A Cisco UC device that advertises itself as a switch router is not considered to be a switch.
  - Voice Service Discovery Protocol (VSDP) is a Cisco Small Business proprietary protocol for SF and SG series managed switches. A Voice Service Discovery Protocol (VSDP) message was received from a neighbor.

In all other cases the operation state is **disabled**.

#### Notes:

- The administrative state cannot be set to **oui-enabled** if the Auto SmartPort Global administrative state is enabled.
- To change the administrative state from **oui-enabled** to **auto-enabled** (or **auto-triggered**), or vice versa, you must first set the administrative state to **disabled**.
- The administrative state cannot be set to **oui-enabled** if the Auto SmartPort administrative state is **enabled**.

#### Examples:

**Example 1** — The following example enables the OUI mode of Voice VLAN. The VLANs 8,9, and 100, which had Smartport ports defined on them, are automatically deactivated.

---

```
Console(config)# voice vlan state oui-enabled
```

---

**Example 2** — The following example disables the Voice VLAN state.

---

```
console(config)#voice vlan state disabled
```

```
All interfaces with Auto Smartport dynamic type will be set to default.
```

```
Are you sure you want to continue? (Y/N)[Y] Y
console(config)#30-Apr-2011 00:04:41 %LINK-W-Down:  Vlan 5
30-Apr-2011 00:04:41 %LINK-W-Down:  Vlan 8
30-Apr-2011 00:04:41 %LINK-W-Down:  Vlan 9
30-Apr-2011 00:04:41 %LINK-W-Down:  Vlan 100
```

---

**Example 3** —The following example sets the Voice VLAN state to auto-enabled. The VLANs are re-activated after the auto SmartPort state is re-applied.

---

```
console(config)#voice vlan state auto-triggered
console(config)#30-Apr-2011 00:13:52 %LINK-I-Up:  Vlan 5
30-Apr-2011 00:13:52 %LINK-I-Up:  Vlan 8
30-Apr-2011 00:13:52 %LINK-I-Up:  Vlan 9
30-Apr-2011 00:13:52 %LINK-I-Up:  Vlan 100
```

---

## 42.2 voice vlan refresh

The **voice vlan refresh** Global Configuration mode command restarts the Voice VLAN discovery process on all the Auto Voice VLAN-enabled switches in the VLAN by removing all externally learned voice VLAN attributes and resetting the voice VLAN to the default voice VLAN.

### Syntax

**voice vlan refresh**

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

Global Configuration mode

---

## Example

```
Console(config)# voice vlan refresh

Console(config)#

30-Apr-2011 02:01:02 %VLAN-I-ReceivedFromVSDP: Voice VLAN updated by VSDP. Voice
VLAN-ID 100, VPT 5, DSCP 46 (Notification that Agreed Voice VLAN is updated)

(Auto Smartport configuration is changed)

30-Apr-2011 02:01:05 %LINK-W-Down: Vlan 50
30-Apr-2011 02:01:05 %LINK-W-Down: Vlan 100
30-Apr-2011 02:01:06 %LINK-I-Up: Vlan 50
30-Apr-2011 02:01:06 %LINK-I-Up: Vlan 100

console#show voice vlan

Administrate Voice VLAN state is auto-triggered

Operational Voice VLAN state is auto-enabled

Best Local Voice VLAN-ID is 100

Best Local VPT is 5 (default)

Best Local DSCP is 46 (default)

(Following is the new active source)

Agreed Voice VLAN is received from switch b0:c6:9a:c1:da:00

Agreed Voice VLAN priority is 2 (active CDP device)

Agreed Voice VLAN-ID is 100

Agreed VPT is 5

Agreed DSCP is 46

Agreed Voice VLAN Last Change is 11-Apr-30 02:01:02
```

---

## 42.3 voice vlan id

Use the **voice vlan id** Global Configuration mode command to statically configure the VLAN identifier of the voice VLAN. The **no** format of the command returns the voice VLAN to the default VLAN (1).

### Syntax

**voice vlan id** *vlan-id*

**no voice vlan id****Parameters**

**vlan id** *vlan-id*—Specifies the voice VLAN (range 1-4094).

**Default Configuration**

VLAN ID 1.

**Command Mode**

Global Configuration mode

**User Guidelines**

If the Voice VLAN does not exist, it is created automatically. It will not be removed automatically by the **no** version of this command.

**Example**

The following example enables VLAN 104 as the voice VLAN on the device.

---

```
Console(config)# voice vlan id 35
```

```
For Auto Voice VLAN, changes in the voice VLAN ID, CoS/802.1p, and/or DSCP will cause the switch to advertise the administrative voice VLAN as static voice VLAN which has higher priority than voice VLAN learnt from external sources.
```

```
Are you sure you want to continue? (Y/N)[Y] Y
```

```
30-Apr-2011 00:19:36 %VLAN-I-VoiceVlanCreated: Voice Vlan ID 104 was created.
```

```
console(config)#30-Apr-2011 00:19:51 %VLAN-I-ReceivedFromVSDP: Voice VLAN updated by VSDP. Voice VLAN-ID 104, VPT 5, DSCP 46
```

---

## 42.4 voice vlan vpt

Use the **voice vlan vpt** Global Configuration mode command to specify a value of VPT (802.1p VLAN priority tag) that will be advertised by LLDP in the Network Policy TLV. The **no** format of the command returns the value to the default.

**Syntax**

**voice vlan vpt** *vpt-value*



**no voice vlan vpt****Parameters**

**vpt** *vpt-value*—The VPT value to be advertised (range 0-7).

**Default Configuration**

5

**Command Mode**

Global Configuration mode

**Example**

The following example sets 7 as the voice VLAN VPT. A notification that the new settings are different than the old ones is displayed.

---

```
Console(config)# voice vlan vpt 7
```

```
For Auto Voice VLAN, changes in the voice VLAN ID, CoS/802.1p, and/or DSCP will cause the switch to advertise the administrative voice VLAN as static voice VLAN which has higher priority than voice VLAN learnt from external sources.
```

```
Are you sure you want to continue? (Y/N)[Y] Y
```

```
30-Apr-2011 00:24:52 %VLAN-W-BestLocal!=Oper: inconsistency detected, VSDP voice VLAN configuration differs from best local. Best local is Voice VLAN-ID 104, VPT 5, DSCP 46
```

```
console(config)#30-Apr-2011 00:25:07 %VLAN-I-ReceivedFromVSDP: Voice VLAN updated by VSDP. Voice VLAN-ID 104, VPT 7, DSCP 46
```

---

## 42.5 voice vlan dscp

Use the **voice vlan dscp** Global Configuration mode command to specify a value of DSCP that will be advertised by LLDP in the Network Policy TLV. The **no** format of the command returns the value to the default.

**Syntax**

**voice vlan dscp** *dscp-value*

**no voice vlan dscp**

### Parameters

**dscp** *dscp-value*—The DSCP value (range 0-63).

### Default Configuration

46

### Command Mode

Global Configuration mode

### Example

The following example sets 63 as the voice VLAN DSCP.

---

```
Console(config)# voice vlan dscp 63
```

```
For Auto Voice VLAN, changes in the voice VLAN ID, CoS/802.1p, and/or DSCP will cause the switch to advertise the administrative voice VLAN as static voice VLAN which has higher priority than voice VLAN learnt from external sources.
```

```
Are you sure you want to continue? (Y/N)[Y] Y
```

```
30-Apr-2011 00:31:07 %VLAN-W-BestLocal!=Oper: inconsistency detected, VSDP voice VLAN configuration differs from best local. Best local is Voice VLAN-ID 104, VPT 7, DSCP 46
```

```
console(config)#30-Apr-2011 00:31:22 %VLAN-I-ReceivedFromVSDP: Voice VLAN updated by VSDP. Voice VLAN-ID 104, VPT 7, DSCP 63
```

---

## 42.6 voice vlan oui-table

Use the **voice vlan oui-table** Global Configuration mode command to configure the voice OUI table. Use the **no** form of this command to restore the default configuration.

### Syntax

**voice vlan oui-table** *{add mac-address-prefix | remove mac-address-prefix}* [*text*]

**no voice vlan oui-table**

### Parameters

- **add** *mac-address-prefix*—Adds the specified MAC address prefix to the voice VLAN OUI table (length: 3 bytes).
- **remove** *mac-address-prefix*—Removes the specified MAC prefix address from the voice VLAN OUI table (length: 3 bytes).
- **text**—Adds the specified text as a description of the specified MAC address to the voice VLAN OUI table (length: 1–32 characters).

### Default Configuration

The default voice VLAN OUI table is:

OUI	Description
00:e0:bb	3COM Phone
00:03:6b	Cisco Phone
00:e0:75	Veritel Polycom Phone
00:d0:1e	Pingtel Phone
00:01:e3	Siemens AG Phone
00:60:b9	NEC/Philips Phone
00:0f:e2	Huawei-3COM Phone
00:09:6e	Avaya Phone

### Command Mode

Global Configuration mode

### User Guidelines

The classification of a packet from VoIP equipment/phones is based on the packet's OUI in the source MAC address. OUIs are globally assigned (administered) by the IEEE.

In MAC addresses, the first three bytes contain a manufacturer ID (Organizationally Unique Identifiers (OUI) ) and the last three bytes contain a unique station ID.

Since the number of IP phone manufacturers that dominates the market is limited and well known, the known OUI values are configured by default and OUIs can be added/removed by the user when required.

### Example

The following example adds an entry to the voice VLAN OUI table.

---

```
Console(config)# voice vlan oui-table add 00:AA:BB description experimental
```

---

## 42.7 voice vlan cos mode

Use the **voice vlan cos mode** Interface Configuration mode command to select the OUI voice VLAN Class of Service (CoS) mode. Use the **no** form of this command to return to the default.

### Syntax

**voice vlan cos mode** *{src / all}*

**no voice vlan cos mode**

### Parameters

- **src**—QoS attributes are applied to packets with OUIs in the source MAC address. See the User Guidelines of [voice vlan oui-table](#).
- **all**—QoS attributes are applied to packets that are classified to the Voice VLAN.

### Default Configuration

The default mode is **src**.

### Command Mode

Global Configuration mode

### Example

The following example applies QoS attributes to voice packets.

---

```
Console(config)# voice vlan cos mode all
```

---

---

## 42.8 voice vlan cos

Use the **voice vlan cos** Global Configuration mode command to set the OUI Voice VLAN Class of Service (CoS). Use the **no** form of this command to restore the default configuration.

### Syntax

**voice vlan *cos* *cos* [*remark*]**

**no voice vlan cos**

### Parameters

- **cos *cos***—Specifies the voice VLAN Class of Service value. (Range: 0–7)
- **remark**—Specifies that the L2 user priority is remarked with the CoS value.

### Default Configuration

The default CoS value is 5.

The L2 user priority is not remarked by default.

### Command Mode

Global Configuration mode

### Example

The following example sets the OUI voice VLAN CoS to 7 and does not do remarking.

---

```
Console(config)# voice vlan cos 7
```

---

## 42.9 voice vlan aging-timeout

Use the **voice vlan aging-timeout** Global Configuration mode command to set the OUI Voice VLAN aging timeout interval. Use the **no** form of this command to restore the default configuration.

### Syntax

**voice vlan aging-timeout *minutes***

## no voice vlan aging-timeout

### Parameters

**aging-timeout** *minutes*—Specifies the voice VLAN aging timeout interval in minutes. (Range: 1–43200).

### Default Configuration

1440 minutes

### Command Mode

Global Configuration mode

### Example

The following example sets the OUI Voice VLAN aging timeout interval to 12 hours.

---

```
Console(config)# voice vlan aging-timeout 720
```

---

## 42.10 voice vlan enable

Use the **voice vlan enable** Interface Configuration (Ethernet, Port-channel) mode command to enable OUI voice VLAN configuration on an interface. Use the **no** form of this command to disable OUI voice VLAN configuration on an interface.

### Syntax

**voice vlan enable**

**no voice vlan enable**

### Default Configuration

Disabled

### Command Mode

Interface Configuration (Ethernet, Port-channel) mode

### User Guidelines

This command is applicable only if the voice VLAN state is configured as OUI voice VLAN.

The port is added to the voice VLAN if a packet with a source MAC address OUI address (defined by [voice vlan oui-table](#)) is trapped on the port. Note: The packet VLAN ID does not have to be the voice VLAN, it can be any VLAN.

The port joins the voice VLAN as a tagged port.

If the time since the last MAC address with a source MAC address OUI address was received on the interface exceeds the timeout limit (configured by [voice vlan aging-timeout](#)), the interface is removed from the voice VLAN.

### Example

The following example enables OUI voice VLAN configuration on `gi2`.

```
Console(config)# interface gi2
Console(config-if)# voice vlan enable
```

## 42.11 show voice vlan

Use the **show voice vlan** EXEC mode command to display the voice VLAN status for all interfaces or for a specific interface if the voice VLAN type is OUI.

### Syntax

```
show voice vlan [type {oui| auto}] [interface-id]
```

### Parameters

- **type oui**—Common and OUI-voice-VLAN specific parameters are displayed.
- **type auto**—Common and Auto Voice VLAN-specific parameters are displayed.
- **interface-id**—Specifies an Ethernet port ID. Relevant only for the OUI type.

### Default Configuration

If the type parameter is omitted the current Voice VLAN type is used.

If interface-id parameter is omitted then information about all interfaces is displayed.

## Command Mode

EXEC mode

## User Guidelines

Using this command without parameters displays the current voice VLAN type parameters and local and agreed voice VLAN settings.

The interface-id parameter is relevant only for the OUI VLAN type.

## Examples:

**Example 1**—Displays the auto voice VLAN parameters.

```
switch>show voice vlan auto

Administrate Voice VLAN state is auto-triggered
Operational Voice VLAN state is auto-enabled
Best Local Voice VLAN-ID is 20
Best Local VPT is 4
Best Local DSCP is 1

Voice VLAN is received from switch 00:01:22:01:ab:87:45
Agreed Voice VLAN priority is 0 (active UC device)
Agreed Voice VLAN-ID is 100
Agreed VPT is 0
Agreed DSCP is 0
Agreed VLAN Last Change is 10-Apr-10 20:01:00
```

**Example 2**—Displays the current voice VLAN parameters.

```
switch>show voice vlan

Administrate Voice VLAN state is auto-enabled
Operational Voice VLAN state is auto-enabled
Best Local Voice VLAN-ID is 1 (default)
Best Local VPT is 0 (default)
Best Local DSCP is 0 (default)
Agreed Voice VLAN is received from switch 00:01:22:01:ab:87:45
Agreed Voice VLAN priority is 2 (static)
```



```
Agreed Voice VLAN-ID is 20
Agreed VPT is 7
Agreed DSCP is 20
Agreed VLAN Last Change is 10-Apr-10 20:01:00
```

---

**Example 3**—Displays the current voice VLAN parameters.

```
switch>show voice vlan
Administrate Voice VLAN state is auto-triggered
Operational Voice VLAN state is disabled
VSDP Authentication is disabled
```

---

**Example 4**—Displays the current voice VLAN parameters.

```
switch>show voice vlan
Administrate Voice VLAN state is disabled
Operational Voice VLAN state is disabled
Best Local Voice VLAN-ID is 20
Best Local VPT is 0 (default)
Best Local DSCP is 0 (default)
Aging timeout: 1440 minutes
CoS: 6
Remark: Yes
```

---

**Example 5**—Displays the voice VLAN parameters when the voice VLAN state is OUI.

```
switch>show voice vlan oui
Administrate Voice VLAN state is oui-enabled
Operational Voice VLAN state is oui-enabled
Best Local Voice VLAN-ID is 1 (default)
Best Local VPT is 4
Best Local DSCP is 1
Aging timeout: 1440 minutes
CoS: 6
```

```

Remark: Yes

OUI table

MAC Address - Prefix      Description
-----
00:E0:BB                  3COM
00:03:6B                  Cisco
00:E0:75                  Veritel
00:D0:1E                  Pingtel
00:01:E3                  Simens
00:60:B9                  NEC/Philips
00:0F:E2                  Huawei-3COM
00:09:6E                  Avaya

Interface      Enabled   Secure   Activated   CoS Mode
-----
gi1            Yes      Yes      Yes         all
gi2            Yes      Yes      No          src
gi3            No       No       -           src

```

## 42.12 show voice vlan local

The **show voice vlan local** EXEC mode command displays information about the auto voice VLAN local configuration, including the best local voice VLAN.

### Syntax

**show voice vlan local**

### Parameters

N/A

### Default Configurations

N/A

**Command Mode**

EXEC mode

**Examples:****Example 1**—A UC is connected to an interface and a conflict is detected:

```
30-Apr-2011 00:39:24 %VLAN-W-ConflictingCDPDetected: conflict detected between
operational VLAN and new CDP device 00:1e:13:73:3d:62 on interface gi7. Platform
TLV is -4FX0-K9, Voice VLAN-ID is 100...
```

---

```
console>show voice vlan local
```

```
Administrate Voice VLAN state is auto-triggered
```

```
Operational Voice VLAN state is auto-enabled
```

```
VSDP Authentication is enabled, key string name is alpha
```

```
The character '*; marks the best local Voice VLAN
```

VLAN-ID	VPT	DSCP	Source	MAC Address	Interface
1	5	46	default	---	---
*104	7	63	static	---	---
100			CDP	00:1e:13:73:3d:62	gi7

---

**Example 2**—Displays the local voice VLAN configuration when the voice VLAN state is Auto-triggered.

```
console>show voice vlan local
```

```
Administrate Voice VLAN state is auto-triggered
```

```
Operational Voice VLAN state is auto-enabled
```

```
VSDP Authentication is enabled, default key chain
```

```
The character '*; marks the best local Voice VLAN
```

VLAN-ID	VPT	DSCP	Source	MAC Address	Interface
1	0	0	default	---	---

---

```

      10  1    27  static      ---      ---
    *10                UC          00:00:12:ea:87:dc  gi1
      10                UC          00:00:aa:aa:89:dc  po1

```

---

**Example 3**—Displays the local voice VLAN configuration when the voice VLAN state is OUI.

```
console>show voice vlan local
```

```
Administrate Voice VLAN state is auto-OUI
```

```
Operational Voice VLAN state is OUI
```

```
The character '*; marks the best local Voice VLAN
```

```

VLAN-ID   VPT   DSCP   Source      MAC Address      Interface
-----
      1    0    0   default     ---             ---
    *10    1    27   static      ---             ---
      10                UC          00:00:12:ea:87:dc  gi1
      10                UC          00:00:aa:aa:89:dc  po1

```

## 43 Smartport Commands

### 43.1 macro auto (Global)

The **macro auto** Global Configuration mode command sets the Auto Smartports administrative global state. The **no** format of the command returns to the default.

#### Syntax

**macro auto** {*enabled* | *disabled* | *controlled*}

**no macro auto**

#### Parameters

- **enabled**—Auto Smartport administrative global and operational states are enabled.
- **disabled**—Auto Smartport administrative global and operational states are disabled.
- **controlled**—Auto Smartport administrative global and operational states are enabled when Auto Voice VLAN is in operation.

#### Default Configuration

Administrative state is **controlled**.

#### Command Mode

Global Configuration mode

#### User Guidelines

Regardless of the status of Auto Smartport, you can always manually apply a Smartport macro to its associated Smartport type. A Smartport macro is either a built-in macro or a user-defined macro. You can define and apply a macro using the CLI commands presented in the Macro Commands section.

If the Auto Smartport Administrative state is controlled, the Auto Smartport Operational state is managed by the Voice VLAN manager and is set as follows:

- Auto Smartport Operational state is disabled when the OUI Voice VLAN is enabled.

- Auto Smartport Operational state is enabled when the Auto Voice VLAN is enabled.

A user cannot enable Auto Smartport globally if the OUI Voice VLAN is enabled.

### Example

This example shows an attempt to enable the Auto Smartport feature globally in the controlled mode. This is not possible because the OUI voice feature is enabled. The voice VLAN state is then disabled, after which Auto Smartports can be enabled. The appropriate VLANs are automatically enabled because the ports are configured for Auto Smartports on these VLANs.

```
console(config)# macro auto controlled
console(config)#macro auto enabled
Auto smartports cannot be enabled because OUI voice is enabled.
console(config)#voice vlan state disabled
console(config)#macro auto enabled
console(config)#10-Apr-2011 16:11:31 %LINK-I-Up:  Vlan 20
10-Apr-2011 16:11:33 %LINK-I-Up:  Vlan 5
10-Apr-2011 16:11:33 %LINK-I-Up:  Vlan 6
10-Apr-2011 16:11:33 %LINK-I-Up:  Vlan 7
10-Apr-2011 16:11:33 %LINK-I-Up:  Vlan 8
10-Apr-2011 16:11:33 %LINK-I-Up:  Vlan 9
10-Apr-2011 16:11:33 %LINK-I-Up:  Vlan 10
```

## 43.2 macro auto smartport (Interface)

The **macro auto smartport** Interface Configuration mode command enables the Auto Smartport feature on a given interface. The **no** format of the command disables the feature on the interface.

### Syntax

**macro auto smartport**

**no macro auto smartport**

**Parameters**

N/A

**Default Configuration**

Enabled.

**Command Mode**

Interface Configuration mode (Ethernet Interface, Port Channel)

**User Guidelines**

This command is effective only when Auto Smartport is globally enabled.

**Example**

Enables the Auto Smartport feature on port 1:

---

```
console(conf)(conf)#interface gi1
console(conf-if)# macro auto smartport
```

---

### 43.3 macro auto trunk refresh

The **macro auto trunk refresh** Global Configuration command reapplies the Smartport macro on a specific interface, or to all the interfaces with the specified Smartport type.

**Syntax**

**macro auto trunk refresh** [*smartport-type*] [*interface-id*]

**Parameters**

- **smartport-type**—Smartport type (switch, router, wireless access point (ap))
- **interface-id**—Interface Identifier (port or port channel).

**Default Configuration**

See User Guidelines.

## Command Mode

Global Configuration mode

## User Guidelines

The **macro auto smartport** command becomes effective only when the Auto Smartport is globally enabled.

If both *smartport-type* and *interface-id* are defined, the attached Smartport macro is executed on the interface if it has the given Smartport type.

If only *smartport-type* is defined, the attached Smartport macro is executed on all interfaces having the given Smartport type.

If only *interface-id* is defined then the corresponding attached Smartport macro is executed if the interface has one of the following Smartport types: **switch**, **router** or wireless access point (**ap**).

If a Smartport macro contains configuration commands that are no longer current on one or more interfaces, you can update their configuration by reapplying the Smartport macro on the interfaces.

## Example

Adds the ports of Smartport type **switch** to all existing VLANs by running the associated Smartport macros.

---

```
console(conf)#macro auto trunk refresh switch
```

---

## 43.4 macro auto resume

The **macro auto resume** Interface Configuration mode command changes the Smartport type from **unknown** to **default** and resumes the Smartport feature on a given interface (but does not reapply the Smartport macro; this is done by [macro auto trunk refresh](#)).

## Syntax

```
macro auto resume
```

## Parameters

N/A



### Default Configuration

N/A

### Command Mode

Interface Configuration mode (Ethernet Interface, Port Channel)

### User Guidelines

When a Smartport macro fails at an interface, the Smartport type of the interface becomes **Unknown**. You must diagnose the reason for the failure on the interface and/or Smartport macro, and correct the error. Before you or Auto Smartport are allowed to reapply the desired Smartport macro, you must reset the interface using the **macro auto resume** command, which changes the Smartport type of the interface to **Default**. Then you can run [macro auto trunk refresh](#).

### Example

Changes the Smartport type from **unknown** to **default** and resumes the Smartport feature on port 1.

---

```
console(conf)interface gi1
console(conf-if)#macro auto resume
```

---

## 43.5 macro auto persistent

The **macro auto persistent** Interface Configuration mode command sets the interface as a Smartport persistent interface. The **no** format of the command returns it to default.

### Syntax

**macro auto persistent**

**no macro auto persistent**

### Parameters

N/A

### Default Configuration

Not persistent.

## Command Mode

Interface Configuration mode (Ethernet Interface, Port Channel)

## User Guidelines

A Smartport's persistent interface retains its dynamic configuration in the following cases: link down/up, the attaching device ages out, and reboot. Note that for persistence and the Smartport configuration to be effective across reboot, the Running Configuration file must be saved to the Startup Configuration file.

## Example

The example establishes two port ranges and makes one persistent and the other not.

```
console(config)#interface range gi1-2
console(config-if-range)#macro auto persistent
console(config-if-range)#exit
console(config)#interface range gi3-4
console(config-if-range)#no macro auto persistent
```

## 43.6 macro auto smartport type

The **macro auto smartport type** Interface Configuration mode command manually (statically) assigns a Smartport type to an interface. The **no** format of the command removes the manually-configured type and returns it to **default**.

### Syntax

```
macro auto smartport type smartport-type [parameter-name value
[parameter-name value [parameter-name value]]]
```

```
no macro auto smartport type
```

### Parameters

**smartport type** *smartport-type*—Smartport type.

*parameter-name value*—Specifies the parameter name and its value (Range: printer, desktop, guest, server, host, ip\_camera, ip\_phone, ip\_phone\_desktop, switch, router or wireless access point (ap)).

## Default Configuration

*parameter-name value*—Parameter default value. For instance, if the parameter is the voice VLAN, the default value is the default voice VLAN.

## Command Mode

Interface Configuration mode (Ethernet Interface, Port Channel)

## User Guidelines

A static type set by the command cannot be changed by a dynamic type.

## Example

This example shows an attempt to set the Smartport type of port 1 to printer (statically). The macro fails at line 10. The **show parser macro name** command is run to display the contents of the macro printer in order to see which line failed.

---

```
console(conf)interface gil
console(conf-if)#macro auto smartport type printer
30-May-2011 15:02:45 %AUTOSMARTPORT-E-FAILEDMACRO: Macro printer for auto smar
port type Printer on interface gil failed at command number 10
console(conf-if)#exit
console(conf-if)#do show parser macro name printer
Macro name : printer
Macro type : default interface
  1. #macro description printer
  2. #macro keywords $native_vlan
  3. #
  4. #macro key description:  $native_vlan: The untag VLAN which will be configu
red on the port
  5. #Default Values are
  6. # $native_vlan = Default VLAN
  7. #
  8. #the port type cannot be detected automatically
  9. #
```

```
10. switchport mode access
11. switchport access vlan $native_vlan
12. #
13. #single host
14. port security max 1
15. port security mode max-addresses
16. port security discard trap 60
17. #
18. smartport storm-control broadcast level 10
19. smartport storm-control include-multicast
20. smartport storm-control broadcast enable
switch030008(config)#
```

---

## 43.7 macro auto processing cdp

The **macro auto processing cdp** Global Configuration mode command enables using CDP capability information to identify the type of an attached device.

When Auto Smartport is enabled on an interface and this command is run, the switch automatically applies the corresponding Smartport type to the interface based on the CDP capabilities advertised by the attaching device(s).

The **no** format of the command disables the feature.

### Syntax

**macro auto processing cdp**

**no macro auto processing cdp**

### Parameters

N/A

### Default Configuration

Enabled

### Command Mode

Global Configuration

**Example**

To enable CDP globally:

---

```
console(conf)#macro auto processing cdp
```

---

**43.8 macro auto processing lldp**

The **macro auto processing lldp** Global Configuration mode command enables using the LLDP capability information to identify the type of an attached device.

When Auto Smartport is enabled on an interface and this command is run, the switch automatically applies the corresponding Smartport type to the interface based on the LLDP capabilities advertised by the attaching device(s).

The **no** format of the command disables the feature.

**Syntax**

**macro auto processing lldp**

**no macro auto processing lldp**

**Parameters**

N/A

**Default Configuration**

Enabled

**Command Mode**

Global Configuration

**Example**

To enable LLDP globally:

---

```
console(conf)#macro auto processing lldp
```

---

---

## 43.9 macro auto processing type

The **macro auto processing type** Global Configuration mode command enables or disables automatic detection of devices of given type. The no format of the command returns to the default.

### Syntax

**macro auto processing type** *smartport-type* {*enabled* | *disabled*}

**no macro auto processing type** *smartport-type*

### Parameters

**smartport-type**—Smartport type (range: host, ip\_phone, ip\_phone\_desktop, switch, router or wireless access point (ap)).

### Default Configuration

By default, auto detection of ip\_phone, ip\_phone\_desktop, switch, and wireless access point (ap) is enabled.

### Command Mode

Global Configuration

### Example

**Example 1** - In this example, VLANs were de-activated because LLDP was disabled.

---

```
console(config)#no macro auto processing lldp
console(config)#10-Apr-2011 16:21:16 %LINK-W-Down:  Vlan 5
10-Apr-2011 16:21:16 %LINK-W-Down:  Vlan 6
10-Apr-2011 16:21:16 %LINK-W-Down:  Vlan 7
10-Apr-2011 16:21:16 %LINK-W-Down:  Vlan 8
10-Apr-2011 16:21:16 %LINK-W-Down:  Vlan 9
10-Apr-2011 16:21:16 %LINK-W-Down:  Vlan 10
10-Apr-2011 16:21:16 %LINK-W-Down:  Vlan 20
```

---

**Example 2** - In this example, automatic detection of wireless access points (ap) is enabled.

```
console(config)#macro auto processing type ?
host                set type to host
ip_phone            set type to ip_phone
ip_phone_desktop    set type to ip_phone_desktop
switch              set type to switch
router              set type to router
ap                  set type to access point
console(config)#macro auto processing type ap enabled
```

## 43.10 macro auto user smartport macro

The **macro auto user smartport macro** Global Configuration mode command links user-defined Smartport macros to a Smartport type. This is done by replacing the link to the built-in macro with the link to the user-defined macro. The **no** format of the command returns the link to the default built-in Smartport macro.

### Syntax

**macro auto user smartport macro** *smartport-type user-defined-macro-name* [*parameter-name value* [*parameter-name value* [*parameter-name value*]]]

**no macro auto user smartport macro** *smartport-type*

### Parameters

- **smartport macro** *smartport-type*—Smartport type (range: printer, desktop, guest, server, host, ip\_camera, ip\_phone, ip\_phone\_desktop, switch, router or wireless access point (ap)).
- **smartport macro** *user-defined-macro-name*—Specifies the user-defined macro name that replaces the built-in Smartport macro.
- *parameter-name value*—Specifies the parameter name and its value in the user-defined macro.

### Default Configuration

**parameter-name** *value*—Parameter's default value. For instance, if the parameter is the native VLAN, the default value is the default native VLAN.

### Command Mode

Global Configuration

### User Guidelines

The scope of each parameter is the macro in which it is defined, with the exception of the parameter **\$voice\_vlan**, which is a global parameter and its value is specified by the switch and cannot be defined in a macro.

The macros must be defined before linking them in this command.

Smartport macros must be disconnected from the Smartport type before removing them (using the **no** version of this command).

To associate a Smartport type with a user-defined macros, you must have defined a pair of macros: one to apply the configuration, and the other (anti macro) to remove the configuration. The macros are paired by their name. The name of the anti macro is the concatenation of **no\_** with the name of the corresponding macro. Please refer to the Macro Command section for details about defining macro.

### Example

To link the user-defined macro: `my_ip_phone_desktop` to the Smartport type: `ip_phone_desktop` and provide values for its two parameters:

---

```
console(conf)#macro auto user smartport macro ip_phone_desktop
my_ip_phone_desktop $p1 1 $p2 2
```

---

## 43.11 macro auto built-in parameters

The **macro auto built-in parameters** Global Configuration mode command replaces the default Auto Smartport values of built-in Smartport macros. The **no** format of the command returns to the default values.

### Syntax

**macro auto built-in parameters** *smartport-type* [*parameter-name value* [*parameter-name value* [*parameter-name value*]]]

**no macro auto built-in parameters** *smartport-type*



## Parameters

**smartport-type**—Smartport type (range: printer, desktop, guest, server, host, ip\_camera, ip\_phone, ip\_phone\_desktop, switch, router or wireless access point (ap)).

**parameter-name value**—Specifies the parameter name and its value. These are the parameters of the built-in or user-defined macro defined in [macro auto user smartport macro](#).

## Default Configuration

The default value of parameter **\$native\_vlan** of the built-in Smartport macros is 1.

For other parameters, the default value is the parameter's default value. For instance, if the parameter is the native VLAN, the default value is the default native VLAN.

## Command Mode

Global Configuration

## User Guidelines

By default, each Smartport type is associated with a pair of built-in macros: a macro that applies the configuration and the anti macro (no macro) to remove the configuration. The Smartport types are the same as the name of the corresponding built-in Smartport macros, with the anti macro prefixed with **no\_**.

The value of the parameter **\$voice\_vlan** cannot be changed by this command.

## Example

To change the parameters of a built-in macro:

---

```
console(conf)#macro auto built-in parameters switch &native_vlan 2
```

---

## 43.12 show macro auto processing

The **show macro auto processing** EXEC mode command displays information about which protocols (CDP/LLDP) are enabled and which device types can be detected automatically.

## Syntax

**show macro auto processing**

**Parameters**

N/A

**Default Configuration**

N/A

**Command Mode**

EXEC

**Example**

```
switch>show macro auto processing
```

```
CDB: enabled
```

```
LLDP: enabled
```

```
host                :disabled
```

```
ip_phone            :enabled
```

```
ip_phone_desktop:enabled
```

```
switch              :enabled
```

```
router              :disabled
```

```
ap                  :enabled
```

---

**43.13 show macro auto smart-macros**

The **show macro auto smart-macros** EXEC mode command displays the name of Smartport macros, their type (built-in or user-defined) and their parameters. This information is displayed for all Smartport types or for the specified one.

**Syntax**

```
show macro auto smart-macros [smartport-type]
```

**Parameters**

**smartport-type**—Smartport type (range: printer, desktop, guest, server, host, ip\_camera, ip\_phone, ip\_phone\_desktop, switch, router or wireless access point (ap)).

### Default Configuration

N/A

### Command Mode

EXEC

### Example

```
switch>show macro auto smartport

Smartport type: ip_phone

Parameters: $native_vlan=1 $voice_vlan=100

Smartport Macro: phone (built-in)

Smartport type: switch

Parameters: $native_vlan =1 $man=aa $enc=qq

Smartport Macro: use-switch

Smartport type: router

Parameters: $native_vlan =2

Smartport Macro: router (built-in)
```

---

## 43.14 show macro auto ports

The **show macro auto ports** EXEC mode command displays information about all Smartport ports or a specific one. If a macro was run on the port and it failed, the type of the port is displayed as Unknown.

### Syntax

```
show macro auto ports [interface-id]
```

### Parameters

**interface-id**—Interface Identifier (Ethernet interface, port channel)

### Default Configuration

Information about all ports is displayed.

### Command Mode

EXEC

## Examples

**Example 1**—Note that Smartport on switch and phone types was configured automatically. Smartport on routers was configured statically.

```
console>show macro auto ports

Smartport is enabled

Administrative Globally Auto Smartport is enabled

Operational Globally Auto Smartport is enabled

Interface      Auto Smartport  Persistent  Smartport Type
              Admin State    State
-----
gi1            disabled        enabled     switch
gi2            enabled         enabled     default
gi3            enabled         disabled    phone
gi4            enabled         enabled     router (static)
gi5            enabled         enabled     switch
gi6            enabled         enabled     unknown
```

**Example 2**—Disabling auto SmartPort on gi2:

```
console(config-if)#interface gi2

console(config-if)#no macro auto smartport

console(config-if)#end

console#show macro auto ports gi2

SmartPort is Enabled

Administrative Globally Auto SmartPort is controlled

Operational Globally Auto SmartPort is enabled

Auto SmartPort is disabled on gi2

Persistent state is not-persistent

Interface type is default

No macro has been activated
```

**Example 3**—Enabling auto Smartport on gi1:

```
console(config-if)#interface gil
console(config-if)#macro auto smartport
console(config-if)#end
console#show macro auto ports gil
SmartPort is Enabled
Administrative Globally Auto SmartPort is enabled
Operational Globally Auto SmartPort is enabled
Auto SmartPort is enabled on gil
Persistent state is persistent
Interface type is switch
Last activated macro is switch
```

---

### 43.15 smartport switchport trunk allowed vlan

The **smartport switchport trunk allowed vlan** Interface Configuration (Ethernet, port-channel) mode command adds/removes VLANs to/from a trunk port.

#### Syntax

```
smartport switchport trunk allowed vlan {add [vlan-list / all] | remove [vlan-list / all]}
```

#### Parameters

**add** *vlan-list* — Specifies a list of VLAN IDs to add to interface. Separate nonconsecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs.

**add** *all* — Add all VLANs to interface.

**remove** *vlan-list* — Specifies a list of VLAN IDs to remove. Separate nonconsecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs.

**remove** *all* — Remove all VLANs from interface.

#### Default Configuration

N/A

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### User Guidelines

This command is an extension of the **switchport trunk allowed vlan** command. Unlike the **switchport trunk allowed vlan** command, the *vlan-list* parameter of this command may include the voice VLAN (when it is the default VLAN). If the default VLAN is the voice VLAN, the following occurs:

- **add all**— Adds the interface to the default VLAN as an egress tagged port.
- **remove all**— Removes the interface from the default VLAN.

### Example

To add port 1 to VLANs 1-5:

---

```
console(conf)#interface gi1
console(conf-if)#smartport switchport trunk allowed vlan add 1-5
```

---

## 43.16 smartport switchport trunk native vlan

Use the **smartport switchport trunk native vlan** Interface Configuration (Ethernet, port-channel) mode command to define the native VLAN when the interface is in trunk mode. Use the **no** form of this command to restore the default configuration.

### Syntax

```
smartport switchport trunk native vlan native-vlan-id
```

### Parameters

**native-vlan-id**—Specifies the native VLAN ID.

### Default Configuration

VLAN 1

### Command Mode

Interface Configuration (Ethernet, port-channel) mode

### User Guidelines

This command is an extension of the **switchport trunk native vlan** CLI command. Unlike the **switchport trunk native vlan** CLI command, this command may also be applied to the default VLAN when the interface belongs to the default VLAN as egress tagged port.

### Example

Define the native VLAN when port 1 is in trunk mode:

---

```
console(conf)interface gi1
console(conf-if)#smartport switchport trunk native vlan 1
```

---

---

## 44 Link Layer Discovery Protocol (LLDP) Commands

---

### 44.1 `lldp run`

Use the `lldp run` Global Configuration mode command to enable LLDP. To disable LLDP, use the `no` form of this command.

#### Syntax

`lldp run`

`no lldp run`

#### Parameters

N/A.

#### Default Configuration

Enabled

#### Command Mode

Global Configuration mode

#### Example

---

```
console(config)# lldp run
```

---

### 44.2 `lldp transmit`

Use the `lldp transmit` Interface Configuration mode command to enable transmitting LLDP on an interface. Use the `no` form of this command to stop transmitting LLDP on an interface.

#### Syntax

`lldp transmit`

`no lldp transmit`

#### Parameters

N/A



### Default Configuration

Enabled

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

LLDP manages LAG ports individually. LLDP sends separate advertisements on each port in a LAG.

LLDP operation on a port is not dependent on the STP state of a port. I.e. LLDP frames are sent on blocked ports.

If a port is controlled by 802.1x, LLDP operates only if the port is authorized.

### Example

```
console(config)# interface gil
console(config-if)# lldp transmit
```

---

## 44.3 Ildp receive

Use the **lldp receive** Interface Configuration mode command to enable receiving LLDP on an interface. Use the **no** form of this command to stop receiving LLDP on an interface.

### Syntax

**lldp receive**

**no lldp receive**

### Parameters

N/A

### Default Configuration

Enabled

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

LLDP manages LAG ports individually. LLDP data received through LAG ports is stored individually per port.

LLDP operation on a port is not dependent on the STP state of a port. I.e. LLDP frames are received on blocked ports.

If a port is controlled by 802.1x, LLDP operates only if the port is authorized.

### Example

```
console(config)# interface gil
console(config-if)# lldp receive
```

## 44.4 lldp timer

Use the **lldp timer** Global Configuration mode command to specify how often the software sends LLDP updates. Use the **no** form of this command to restore the default configuration.

### Syntax

**lldp timer** *seconds*

**no lldp timer**

### Parameters

**timer** *seconds*—Specifies, in seconds, how often the software sends LLDP updates (range: 5-32768 seconds).

### Default Configuration

30 seconds.

### Command Mode

Global Configuration mode

### Example

The following example sets the interval for sending LLDP updates to 60 seconds.

```
Console(config)# lldp timer 60
```

---

## 44.5 lldp hold-multiplier

Use the **lldp hold-multiplier** Global Configuration mode command to specify how long the receiving device holds a LLDP packet before discarding it. Use the **no** form of this command to restore the default configuration.

### Syntax

**lldp hold-multiplier** *number*

**no lldp hold-multiplier**

### Parameters

**hold-multiplier** *number*—Specifies the LLDP packet hold time interval as a multiple of the LLDP timer value (range: 2-10).

### Default Configuration

The default LLDP hold multiplier is 4.

### Command Mode

Global Configuration mode

### User Guidelines

The actual Time-To-Live (TTL) value of LLDP frames is calculated by the following formula:

$$\text{TTL} = \min(65535, \text{LLDP-Timer} * \text{LLDP-hold-multiplier})$$

For example, if the value of the LLDP timer is 30 seconds, and the value of the LLDP hold multiplier is 4, then the value 120 is encoded in the TTL field of the LLDP header.

### Example

The following example sets the LLDP packet hold time interval to 90 seconds.

---

```
Console(config)# lldp timer 30
Console(config)# lldp hold-multiplier 3
```

---

## 44.6 lldp reinit

Use the **lldp reinit** Global Configuration mode command to specify the minimum time an LLDP port waits before reinitializing LLDP transmission. Use the **no** form of this command to revert to the default setting.

### Syntax

**lldp reinit** *seconds*

**no lldp reinit**

### Parameters

**reinit** *seconds*—Specifies the minimum time in seconds an LLDP port waits before reinitializing LLDP transmission.(Range: 1–10)

### Default Configuration

2 seconds

### Command Mode

Global Configuration mode

### Example

---

```
console(config)# lldp reinit 4
```

---

## 44.7 lldp tx-delay

Use the **lldp tx-delay** Global Configuration mode command to set the delay between successive LLDP frame transmissions initiated by value/status changes in the LLDP local systems MIB. Use the **no** form of this command to restore the default configuration.

### Syntax

**lldp tx-delay** *seconds*

**no lldp tx-delay**

### Parameters

**tx-delay *seconds***—Specifies the delay in seconds between successive LLDP frame transmissions initiated by value/status changes in the LLDP local systems MIB (range: 1-8192 seconds).

### Default Configuration

The default LLDP frame transmission delay is 2 seconds.

### Command Mode

Global Configuration mode

### User Guidelines

It is recommended that the tx-delay be less than 0.25 of the LLDP timer interval.

### Example

The following example sets the LLDP transmission delay to 10 seconds.

---

```
Console(config)# lldp tx-delay 10
```

---

## 44.8 lldp optional-tlv

Use the **lldp optional-tlv** Interface Configuration (Ethernet) mode command to specify which optional TLVs are transmitted. Use the **no** form of this command to restore the default configuration.

### Syntax

```
lldp optional-tlv {tlv [tlv2 ... tlv5] | none}
```

```
no lldp optional-tlv
```

### Parameters

**tlv**—Specifies TLV to be included. Available optional TLVs are: port-desc, sys-name, sys-desc, sys-cap, 802.3-mac-phy, 802.3-lag, 802.3-max-frame-size.

**none** - No optional TLV is transmitted

### Default Configuration

The syscap is transmitted

## Command Mode

Interface Configuration (Ethernet) mode

## Example

The following example specifies that the port description TLV is transmitted on gigabitethernet port 2.

```
Console(config)# interface gi2
Console(config-if)# lldp optional-tlv port-desc
```

---

## 44.9 lldp management-address

Use the **lldp management-address** Interface Configuration (Ethernet) mode command to specify the management address advertised by an interface. Use the **no** form of this command to stop advertising management address information.

## Syntax

**lldp management-address** *{ip-address / none / automatic [interface-id]}*

**no lldp management-address**

## Parameters

- **ip-address**—Specifies the static management address to advertise.
- **none**—Specifies that no address is advertised.
- **automatic**—Specifies that the software automatically selects a management address to advertise from all the IP addresses of the product. In case of multiple IP addresses, the software selects the lowest IP address among the dynamic IP addresses. If there are no dynamic addresses, the software selects the lowest IP address among the static IP addresses.
- **automatic interface-id**—(Available only when the device is in Layer 3 (router mode)). Specifies that the software automatically selects a management address to advertise from the IP addresses that are configured on the interface ID. In case of multiple IP addresses, the software selects the lowest IP address among the dynamic IP addresses of the interface. If there are no dynamic addresses, the software selects the lowest IP address among the static IP addresses of the interface. The interface ID can be one of the following types: Ethernet port, port-channel or VLAN. Note that if the

port or port- channel are members in a VLAN that has an IP address, that address is not included because the address is associated with the VLAN.

### Default Configuration

No IP address is advertised.

The default advertisement is **automatic**.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

Each port can advertise one IP address.

### Example

The following example sets the LLDP management address advertisement mode to **automatic** on `gi2`.

```
Console(config)# interface gi2
Console(config-if)# lldp management-address automatic
```

## 44.10 lldp notifications

Use the **lldp notifications** Interface Configuration (Ethernet) mode command to enable/disable sending LLDP notifications on an interface. Use the **no** form of this command to restore the default configuration.

### Syntax

**lldp notifications** *{enable / disable}*

**no lldp notifications**

### Parameters

- **enable**—Enables sending LLDP notifications.
- **disable**—Disables sending LLDP notifications.

### Default Configuration

Disabled.

### Command Mode

Interface Configuration (Ethernet) mode

### Example

The following example enables sending LLDP notifications on `gi5`.

---

```
Console(config)# interface gi5
Console(config-if)# lldp notifications enable
```

---

## 44.11 lldp notifications interval

Use the **lldp notifications interval** Global Configuration mode command to configure the maximum transmission rate of LLDP notifications. Use the **no** form of this command to return to the default.

### Syntax

**lldp notifications interval** *seconds*

**no lldp notifications interval**

### Parameters

**interval** *seconds*—The device does not send more than a single notification in the indicated period (range: 5–3600).

### Default Configuration

5 seconds

### Command Mode

Global Configuration mode

### Example

---

```
console(config)# lldp notifications interval 10
```

---



## 44.12 Ildp Ildpdu

The `ldp lldpdu` Global Configuration mode command defines LLDP packet handling when LLDP is globally disabled. To restore the default configuration, use the `no` form of this command.

### Syntax

`ldp lldpdu { filtering | flooding }`

`no ldp lldpdu`

### Parameters

- **filtering** — Specifies that when LLDP is globally disabled, LLDP packets are filtered (deleted).
- **flooding** — Specifies that when LLDP is globally disabled, LLDP packets are flooded (forwarded to all interfaces).

### Default Configuration

LLDP packets are filtered when LLDP is globally disabled.

### Command Mode

Global Configuration mode

### User Guidelines

If the STP mode is MSTP, the LLDP packet handling mode cannot be set to **flooding**.

The STP mode cannot be set to MSTP if the LLDP packet handling mode is **flooding**.

If LLDP is globally disabled, and the LLDP packet handling mode is **flooding**, LLDP packets are treated as data packets with the following exceptions:

- VLAN ingress rules are not applied to LLDP packets. The LLDP packets are trapped on all ports for which the STP state is Forwarding.
- Default "deny-all" rules are not applied to LLDP packets.
- VLAN egress rules are not applied to LLDP packets. The LLDP packets are flooded to all ports for which the STP state is Forwarding.
- LLDP packets are sent as untagged.

### Example

The following example sets the LLDP packet handling mode to Flooding when LLDP is globally disabled.

```
Console(config)# lldp lldpdu flooding
```

## 44.13 lldp med

Use the **lldp med** Interface Configuration (Ethernet) mode command to enable or disable LLDP Media Endpoint Discovery (MED) on a port. Use the **no** form of this command to return to the default state.

### Syntax

```
lldp med {enable [tlv... tlv4] | disable}
```

```
no lldp med
```

### Parameters

**enable** - Enable LLDP MED

**tlv**—Specifies the TLV that should be included. Available TLVs are: network-policy, location, and poe-pse, inventory. The capabilities TLV is always included if LLDP-MED is enabled.

**disable** - disable LLDP MED on the port

### Default Configuration

Enabled with network-policy TLV

### Command Mode

Interface Configuration (Ethernet) mode

### Example

The following example enables LLDP MED with the **location** TLV on **gi3**.

```
Console(config)# interface gi3  
Console(config-if)# lldp med enable location
```

---

### 44.14 lldp med notifications topology-change

Use the **lldp med notifications topology-change** Interface Configuration (Ethernet) mode command to enable sending LLDP MED topology change notifications on a port. Use the **no** form of this command to restore the default configuration.

#### Syntax

**lldp med notifications topology-change** *{enable / disable}*

**no lldp med notifications topology-change**

#### Parameters

- **enable**—Enables sending LLDP MED topology change notifications.
- **disable**—Disables sending LLDP MED topology change notifications.

#### Default Configuration

Disable is the default.

#### Command Mode

Interface Configuration (Ethernet) mode

#### Example

The following example enables sending LLDP MED topology change notifications on `gi2`.

---

```
Console(config)# interface gi2
Console(config-if)# lldp med notifications topology-change enable
```

---

### 44.15 lldp med fast-start repeat-count

When a port comes up, LLDP can send packets more quickly than usual using its fast-start mechanism.

Use the **lldp med fast-start repeat-count** Global Configuration mode command to configure the number of packets that is sent during the activation of the fast start mechanism. Use the **no** form of this command return to default.

### Syntax

**lldp med fast-start repeat-count** *number*

**no lldp med fast-start repeat-count**

### Parameters

**repeat-count** *number*—Specifies the number of times the fast start LLDPDU is being sent during the activation of the fast start mechanism. The range is 1-10.

### Default Configuration

3

### Command Mode

Global Configuration mode

### Example

```
console(config)# lldp med fast-start repeat-count 4
```

---

## 44.16 lldp med network-policy (global)

Use the **lldp med network-policy** Global Configuration mode command to define a LLDP MED network policy. For voice applications, it is simpler to use **lldp med network-policy voice auto**.

The **lldp med network-policy** command creates the network policy, which is attached to a port by **lldp med network-policy (interface)**.

The network policy defines how LLDP packets are constructed.

Use the **no** form of this command to remove LLDP MED network policy.

### Syntax

**lldp med network-policy** *number application [vlan vlan-id] [vlan-type {tagged / untagged}] [up priority] [dscp value]*

**no lldp med network-policy** *number*

### Parameters

- **number**—Network policy sequential number. The range is 1-32.

- **application**—The name or the number of the primary function of the application defined for this network policy. Available application names are:
  - voice
  - voice-signaling
  - guest-voice
  - guest-voice-signaling
  - softphone-voice
  - video-conferencing
  - streaming-video
  - video-signaling.
- **vlan *vlan-id***—VLAN identifier for the application.
- **vlan-type**—Specifies if the application is using a tagged or an untagged VLAN.
- **up *priority***—User Priority (Layer 2 priority) to be used for the specified application.
- **dscp *value***—DSCP value to be used for the specified application.

### Default Configuration

No network policy is defined.

### Command Mode

Global Configuration mode

### User Guidelines

Use the **lldp med network-policy** Interface Configuration command to attach a network policy to a port.

Up to 32 network policies can be defined.

### Example

This example creates a network policy for the voice-signally application and attaches it to port 1. LLDP packets sent on port 1 will contain the information defined in the network policy.

```
console(config)# lldp med network-policy 1 voice-signaling vlan 1 vlan-type
untagged up 1 dscp 2

Console(config)# interface gil

Console(config-if)# lldp med network-policy add 1
```

## 44.17 lldp med network-policy (interface)

Use the **lldp med network-policy** Interface Configuration (Ethernet) mode command to attach or remove an LLDP MED network policy on a port. Network policies are created in [lldp med network-policy \(global\)](#).

Use the **no** form of this command to remove all the LLDP MED network policies from the port.

### Syntax

**lldp med network-policy** *{add / remove}* *number*

**no lldp med network-policy** *number*

### Parameters

- **number**—Specifies the network policy sequential number. The range is 1-32
- **add/remove number**—Attaches/removes the specified network policy to the interface.

### Default Configuration

No network policy is attached to the interface.

### Command Mode

Interface Configuration (Ethernet) mode

### User Guidelines

For each port, only one network policy per application (voice, voice-signaling, etc.) can be defined.

### Example

This example creates a network policy for the voice-signally application and attaches it to port 1. LLDP packets sent on port 1 will contain the information defined in the network policy.

```
console(config)# lldp med network-policy 1 voice-signaling vlan 1 vlan-type
untagged up 1 dscp 2

Console(config)# interface gi1

Console(config-if)# lldp med network-policy add 1
```

## 44.18 lldp med network-policy voice auto

A network policy for voice LLDP packets can be created by using the [lldp med network-policy \(global\)](#). The **lldp med network-policy voice auto** Global Configuration mode is simpler in that it uses the configuration of the Voice application to create the network policy instead of the user having to manually configure it.

The **lldp med network-policy voice auto** command generates an LLDP MED network policy for voice, if the voice VLAN operation mode is **auto voice VLAN**. The voice VLAN, 802.1p priority, and the DSCP of the voice VLAN are used in the policy. Use the **no** form of this command to disable this mode. The network policy is attached automatically to the voice VLAN.

### Syntax

```
lldp med network-policy voice auto
```

```
no lldp med network-policy voice auto
```

### Parameters

N/A

### Default Configuration

N/A

### Command Mode

Global Configuration mode

### User Guidelines

In Auto mode, the Voice VLAN feature determines on which interfaces to advertise the network policy TLV with application type **voice**, and controls the parameters of that TLV.

To enable the auto generation of a network policy based on the auto voice VLAN, there must be no manual pre-configured network policies for the voice application

In Auto mode, you cannot manually define a network policy for the voice application using the `lldp med network-policy (global)` command.

### Example

```
console(config)# lldp med network-policy voice auto
```

---

## 44.19 clear lldp table

Use the **clear lldp table** command in Privileged EXEC mode to clear the neighbors table for all ports or for a specific port.

### Syntax

```
clear lldp table [interface-id]
```

### Parameters

**interface-id**—Specifies a port ID.

### Default Configuration

If no interface is specified, the default is to clear the LLDP table for all ports.

### Command Mode

Privileged EXEC mode

### Example

```
console# clear lldp table gil
```



## 44.20 lldp med location

Use the **lldp med location** Interface Configuration (Ethernet) mode command to configure the location information for the LLDP Media Endpoint Discovery (MED) for a port. Use the **no** form of this command to delete location information for a port.

### Syntax

```
lldp med location {{coordinate data} | {civic-address data} | {ecs-elin data}}
```

```
no lldp med location {coordinate | civic-address | ecs-elin}
```

### Parameters

- **coordinate data**—Specifies the location data as coordinates in hexadecimal format.
- **civic-address data**—Specifies the location data as a civic address in hexadecimal format.
- **ecs-elin data**—Specifies the location data as an Emergency Call Service Emergency Location Identification Number (ECS ELIN) in hexadecimal format.
- **data**—Specifies the location data in the format defined in ANSI/TIA 1057: dotted hexadecimal data: Each byte in a hexadecimal character string is two hexadecimal digits. Bytes are separated by a period or colon. (Length: coordinate: 16 bytes. Civic-address: 6-160 bytes. Ecs-elin: 10-25 bytes)

### Default Configuration

The location is not configured.

### Command Mode

Interface Configuration (Ethernet) mode

### Example

The following example configures the LLDP MED location information on `gi2` as a civic address.

```
console(config)# interface gi2
console(config-if)# lldp med location civic-address 616263646566
```

## 44.21 show lldp configuration

Use the **show lldp configuration** Privileged EXEC mode command to display the LLDP configuration for all ports or for a specific port.

### Syntax

**show lldp configuration** [*interface-id*]

### Parameters

**interface-id**—Specifies the port ID.

### Default Configuration

N/A

### Command Mode

Privileged EXEC mode

### Examples

**Example 1** - Display LLDP configuration for all ports.

```
Switch# show lldp configuration
```

```
State: Enabled
```

```
Timer: 30 Seconds
```

```
Hold multiplier: 4
```

```
Reinit delay: 2 Seconds
```

```
Tx delay: 2 Seconds
```

```
Notifications interval: 5 seconds
```

```
LLDP packets handling: Filtering
```

Port	State	Optional TLVs	Address	Notifications
gi1	RX,TX	PD, SN, SD, SC	172.16.1.1	Disabled
gi2	TX	PD, SN	172.16.1.1	Disabled
gi3	RX,TX	PD, SN, SD, SC	None	Disabled

```

gi5      RX,TX  D,  SN, SD, SC      automatic  Disabled
gi6      RX,TX  PD, SN, SD, SC      auto vlan 1 Disabled
gi7      RX,TX  PD, SN, SD, SC      auto gl    Disabled
gi8      RX,TX  PD, SN, SD, SC      auto chl   Disabled

```

### Example 2 - Display LLDP configuration for port 1.

```

Switch# show lldp configuration gi1

State: Enabled
Timer: 30 Seconds
Hold multiplier: 4
Reinit delay: 2 Seconds
Tx delay: 2 Seconds
Notifications interval: 5 seconds
LLDP packets handling: Filtering

Port State   Optional TLVs   Address   Notifications
-----
gi1 RX, TX   PD, SN, SD, SC  72.16.1.1  Disabled

802.3 optional TLVs: 802.3-mac-phy, 802.3-lag, 802.3-max-frame-size
802.1 optional TLVs
PVID: Enabled
PPVIDs: 0, 1, 92
VLANs: 1, 92
Protocols: 802.1x

```

The following table describes the significant fields shown in the display:

Field	Description
<b>Timer</b>	The time interval between LLDP updates.
<b>Hold multiplier</b>	The amount of time (as a multiple of the timer interval) that the receiving device holds a LLDP packet before discarding it.
<b>Reinit timer</b>	The minimum time interval an LLDP port waits before re-initializing an LLDP transmission.
<b>Tx delay</b>	The delay between successive LLDP frame transmissions initiated by value/status changes in the LLDP local systems MIB.
<b>Port</b>	The port number.
<b>State</b>	The port's LLDP state.
<b>Optional TLVs</b>	Optional TLVs that are advertised. Possible values are: PD - Port description SN - System name SD - System description SC - System capabilities
<b>Address</b>	The management address that is advertised.
<b>Notifications</b>	Indicates whether LLDP notifications are enabled or disabled.
<b>PVID</b>	Port VLAN ID advertised.
<b>PPVID</b>	Protocol Port VLAN ID advertised.
<b>Protocols</b>	Protocols advertised.

## 44.22 show lldp med configuration

Use the **show lldp med configuration** Privileged EXEC mode command to display the LLDP Media Endpoint Discovery (MED) configuration for all ports or for a specific port.

### Syntax

**show lldp med configuration** *[interface-id]*

### Parameters

**interface-id**—Specifies a port ID.

**Default Configuration**

If no port ID is entered, the command displays information for all ports.

**Command Mode**

Privileged EXEC mode

**Examples**

**Example 1** - The following example displays the LLDP MED configuration for all interfaces.

---

```

console# show lldp med configuration

Fast Start Repeat Count: 4.

lldp med network-policy voice: manual

Network policy 1
-----

Application type: voiceSignaling
VLAN ID: 1 untagged
Layer 2 priority: 0
DSCP: 0

Port      Capabilities  Network Policy Location  Notifications  Inventory
-----  -
gi1      Yes           Yes       Yes      Enabled       Yes
gi2      Yes           Yes       No       Enabled       No
gi3      No            No        No       Enabled       No

```

---

**Example 2** - The following example displays the LLDP MED configuration for gi1.

---

```

console# show lldp med configuration gi1

Port      Capabilities  Network Policy Location  Notifications  Inventory
-----  -
gi1      Yes           Yes       Yes      Enabled       Yes

```

---

```

Network policies:
Location:
Civic-address: 61:62:63:64:65:66

```

---

## 44.23 show lldp local tlvs-overloading

When an LLDP packet contains too much information for one packet, this is called overloading. Use the **show lldp local tlvs-overloading** EXEC mode command to display the status of TLVs overloading of the LLDP on all ports or on a specific port.

### Syntax

```
show lldp local tlvs-overloading [interface-id]
```

### Parameters

**interface-id**—Specifies a port ID.

### Default Configuration

If no port ID is entered, the command displays information for all ports.

### Command Mode

EXEC mode

### User Guidelines

The command calculates the overloading status of the current LLDP configuration, and not for the last LLDP packet that was sent.

### Example

---

```

Switch# show lldp local tlvs-overloading gil

```

TLVs Group	Bytes	Status
-----	-----	-----
Mandatory	31	Transmitted
LLDP-MED Capabilities	9	Transmitted
LLDP-MED Location	200	Transmitted
802.1	1360	Overloading

Total: 1600 bytes

Left: 100 bytes

---

## 44.24 show lldp local

Use the **show lldp local** Privileged EXEC mode command to display the LLDP information that is advertised from a specific port.

### Syntax

**show lldp local** *interface-id*

### Parameters

**Interface-id**—Specifies a port ID.

### Default Configuration

If no port ID is entered, the command displays information for all ports.

### Command Mode

Privileged EXEC mode

### Example

The following examples display LLDP information that is advertised from `gi 1` and `2`.

---

```
Switch# show lldp local gi1
Device ID: 0060.704C.73FF
Port ID: gi1
Capabilities: Bridge
System Name: ts-7800-1
System description:
Port description:
Management address: 172.16.1.8
802.3 MAC/PHY Configuration/Status
Auto-negotiation support: Supported
```

```
Auto-negotiation status: Enabled
Auto-negotiation Advertised Capabilities: 100BASE-TX full duplex, 1000BASE-T full duplex
Operational MAU type: 1000BaseTFD
802.3 Link Aggregation
Aggregation capability: Capable of being aggregated
Aggregation status: Not currently in aggregation
Aggregation port ID: 1
802.3 Maximum Frame Size: 1522

802.3 EEE
Local Tx: 30 usec
Local Rx: 25 usec
Remote Tx Echo: 30 usec
Remote Rx Echo: 25 usec

802.1 PVID: 1
802.1 PPVID: 2 supported, enabled
802.1 VLAN: 2 (VLAN2)
802.1 Protocol: 88 8E 01

LLDP-MED capabilities: Network Policy, Location Identification
LLDP-MED Device type: Network Connectivity
LLDP-MED Network policy
Application type: Voice
Flags: Tagged VLAN
VLAN ID: 2
Layer 2 priority: 0
DSCP: 0

LLDP-MED Power over Ethernet
```



```
Device Type: Power Sourcing Entity
Power source: Primary Power Source
Power priority: High
Power value: 9.6 Watts
```

#### LLDP-MED Location

```
Coordinates: 54:53:c1:f7:51:57:50:ba:5b:97:27:80:00:00:67:01
Hardware Revision: B1
Firmware Revision: A1
Software Revision: 3.8
Serial number: 7978399
Manufacturer name: Manufacturer
Model name: Model 1
Asset ID: Asset 123
Switch# show lldp local gi2
LLDP is disabled.
```

---

## 44.25 show lldp statistics

Use the **show lldp statistics** EXEC mode command to display LLDP statistics on all ports or a specific port.

### Syntax

```
show lldp statistics [interface-id]
```

### Parameters

**interface-id**—Specifies a port ID.

### Default Configuration

If no port ID is entered, the command displays information for all ports.

### Command Mode

EXEC mode

**Example**

```
Switch# show lldp statistics
```

```
console(config-if)# do show lldp statistics
```

```
Tables Last Change Time: 14-Oct-2010 32:08:18
```

```
Tables Inserts: 26
```

```
Tables Deletes: 2
```

```
Tables Dropped: 0
```

```
Tables Ageouts: 1
```

Port	TX Frames		RX Frame		RX TLVs		RX Ageouts
	Total	Total	Discarded	Errors	Discarded	Unrecognized	Total
gi1	730	850	0	0	0	0	0
gi2	0	0	0	0	0	0	0
gi3	730	0	0	0	0	0	0
gi4	0	0	0	0	0	0	0
gi5	0	0	0	0	0	0	0
gi6	8	7	0	0	0	0	1
gi7	0	0	0	0	0	0	0
gi8	0	0	0	0	0	0	0
gi9	730	0	0	0	0	0	0
gi10	0	0	0	0	0	0	0

**44.26 show lldp neighbors**

Use the **show lldp neighbors** Privileged EXEC mode command to display information about neighboring devices discovered using LLDP. The information can be displayed for all ports or for a specific port.

**Syntax**

```
show lldp neighbors [interface-id][detail | secondary]
```

**Parameters**

**interface-id**—Specifies a port ID.

**detail**—Displays detailed information about a neighbor (or neighbors) from the main cache.

**secondary**—Displays information about neighbors from the secondary cache.

**Default Configuration**

If no port ID is entered, the command displays information for all ports.

Detail is the default parameter.

**Command Mode**

Privileged EXEC mode

**User Guidelines**

A TLV value that cannot be displayed as an ASCII string is displayed as an hexadecimal string.

**Examples**

**Example 1** - The following example displays information about neighboring devices discovered using LLDP on all ports.

Location information, if it exists, is also displayed.

---

```
Switch# show lldp neighbors
```

Port	Device ID	Port ID	System Name	Capabilities	TTL
gi1	00:00:00:11:11:11	gi1	ts-7800-2	B	90
gi1	00:00:00:11:11:11 D	gi1	ts-7800-2	B	90
gi2	00:00:26:08:13:24	gi3	ts-7900-1	B, R	90
gi3	00:00:26:08:13:24	gi2	ts-7900-2	W	90

---

**Example 2** - The following example displays information about neighboring devices discovered using LLDP port 1.

```
Switch# show lldp neighbors gi1

Device ID: 00:00:00:11:11:11
Port ID: gi
System Name: ts-7800-2
Capabilities: B
System description:
Port description:
Management address: 172.16.1.1
Time To Live: 90 seconds

802.3 MAC/PHY Configuration/Status
Auto-negotiation support: Supported.
Auto-negotiation status: Enabled.
Auto-negotiation Advertised Capabilities: 100BASE-TX full duplex, 1000BASE-T full duplex.
Operational MAU type: 1000BaseTFD

802.3 Power via MDI
MDI Power support Port Class: PD
PSE MDI Power Support: Not Supported
PSE MDI Power State: Not Enabled
PSE power pair control ability: Not supported.
PSE Power Pair: Signal
PSE Power class: 1

802.3 Link Aggregation
Aggregation capability: Capable of being aggregated
```

Aggregation status: Not currently in aggregation

Aggregation port ID: 1

802.3 Maximum Frame Size: 1522

802.3 EEE

Remote Tx: 25 usec

Remote Rx: 30 usec

Local Tx Echo: 30 usec

Local Rx Echo: 25 usec

802.1 PVID: 1

802.1 PPVID: 2 supported, enabled

802.1 VLAN: 2(VLAN2)

802.1 Protocol: 88 8E 01

LLDP-MED capabilities: Network Policy.

LLDP-MED Device type: Endpoint class 2.

LLDP-MED Network policy

Application type: Voice

Flags: Unknown policy

VLAN ID: 0

Layer 2 priority: 0

DSCP: 0

LLDP-MED Power over Ethernet

Device Type: Power Device

Power source: Primary power

Power priority: High

Power value: 9.6 Watts

```

Hardware revision: 2.1
Firmware revision: 2.3
Software revision: 2.7.1
Serial number: LM759846587
Manufacturer name: VP
Model name: TR12
Asset ID: 9

```

LLDP-MED Location

```
Coordinates: 54:53:c1:f7:51:57:50:ba:5b:97:27:80:00:00:67:01
```

The following table describes significant LLDP fields shown in the display:

Field	Description
<b>Port</b>	The port number.
<b>Device ID</b>	The neighbor device's configured ID (name) or MAC address.
<b>Port ID</b>	The neighbor device's port ID.
<b>System name</b>	The neighbor device's administratively assigned name.
<b>Capabilities</b>	The capabilities discovered on the neighbor device. Possible values are:  B - Bridge R - Router W - WLAN Access Point T - Telephone D - DOCSIS cable device H - Host r - Repeater O - Other
<b>System description</b>	The neighbor device's system description.
<b>Port description</b>	The neighbor device's port description.

Field	Description
<b>Management address</b>	The neighbor device's management address.
<b>Auto-negotiation support</b>	The auto-negotiation support status on the port. (Supported or Not Supported)
<b>Auto-negotiation status</b>	The active status of auto-negotiation on the port. (Enabled or Disabled)
<b>Auto-negotiation Advertised Capabilities</b>	The port speed/duplex/flow-control capabilities advertised by the auto-negotiation.
<b>Operational MAU type</b>	The port MAU type.
<b>LLDP MED</b>	
<b>Capabilities</b>	The sender's LLDP-MED capabilities.
<b>Device type</b>	The device type. Indicates whether the sender is a Network Connectivity Device or Endpoint Device, and if an Endpoint, to which Endpoint Class it belongs.
<b>LLDP MED - Network Policy</b>	
<b>Application type</b>	The primary function of the application defined for this network policy.
<b>Flags</b>	Flags. The possible values are:  Unknown policy: Policy is required by the device, but is currently unknown.  Tagged VLAN: The specified application type is using a Tagged VLAN.  Untagged VLAN: The specified application type is using an Untagged VLAN.
<b>VLAN ID</b>	The VLAN identifier for the application.
<b>Layer 2 priority</b>	The Layer 2 priority used for the specified application.
<b>DSCP</b>	The DSCP value used for the specified application.
<b>LLDP MED - Power Over Ethernet</b>	
<b>Power type</b>	The device power type. The possible values are: Power Sourcing Entity (PSE) or Power Device (PD).

Field	Description
<b>Power Source</b>	The power source utilized by a PSE or PD device. A PSE device advertises its power capability. The possible values are: Primary power source and Backup power source. A PD device advertises its power source. The possible values are: Primary power, Local power, Primary and Local power.
<b>Power priority</b>	The PD device priority. A PSE device advertises the power priority configured for the port. A PD device advertises the power priority configured for the device. The possible values are: Critical, High and Low.
<b>Power value</b>	The total power in watts required by a PD device from a PSE device, or the total power a PSE device is capable of sourcing over a maximum length cable based on its current configuration.
<b>LLDP MED - Location</b>	
<b>Coordinates, Civic address, ECS ELIN.</b>	The location information raw data.



## 45 CDP Commands

### 45.1 cdp run

The **cdp run** Global Configuration mode command enables CDP globally. The **no** format of this command disabled CDP globally.

#### Syntax

**cdp run**

**no cdp run**

#### Parameters

N/A

#### Default Configuration

Enabled.

#### Command Mode

Global Configuration mode

#### User Guidelines

CDP is a link layer protocols for directly-connected CDP/LLDP-capable devices to advertise themselves and their capabilities. In deployments where the CDP/LLDP capable devices are not directly connected and are separated with CDP/LLDP incapable devices, the CDP/LLDP capable devices may be able to receive the advertisement from other device(s) only if the CDP/LLDP incapable devices flood the CDP/LLDP packets they receives. If the CDP/LLDP incapable devices perform VLAN-aware flooding, then CDP/LLDP capable devices can hear each other only if they are in the same VLAN. It should be noted that a CDP/LLDP capable device may receive advertisement from more than one device if the CDP/LLDP incapable devices flood the CDP/LLDP packets.

To learn and advertise CDP information, it must be globally enabled (it is so by default) and also enabled on interfaces (also by default).

#### Example

```
console(conf) cdp run
```

---

## 45.2 cdp enable

The **cdp enable** Interface Configuration mode command enables CDP on interface. The **no** format of the CLI command disables CDP on an interface.

### Syntax

**cdp enable**

### Parameters

N/A

### Default Configuration

Enabled

### Command Mode

Ethernet Interface

### User Guidelines

For CDP to be enabled on an interface, it must first be enabled globally using [cdp run](#).

### Example

---

```
console(conf) cdp run
console(conf) interface gil
console(conf-if) cdp enable
```

---

## 45.3 cdp pdu

Use the **cdp pdu** Global Configuration mode command when CDP is not enabled globally. It specifies CDP packets handling when CDP is globally disabled. The **no** format of this command returns to default.

### Syntax

**cdp pdu [filtering | bridging | flooding]**

**no cdp pdu**

### Parameters

**filtering**—Specify that when CDP is globally disabled, CDP packets are filtered (deleted).

**bridging**—Specify that when CDP is globally disabled, CDP packets are bridged as regular data packets (forwarded based on VLAN).

**flooding**—Specify that when CDP is globally disabled, CDP packets are flooded to all the ports in the product that are in STP forwarding state, ignoring the VLAN filtering rules.

### Default Configuration

**bridging**

### Command Mode

Global Configuration mode

### User Guidelines

When CDP is globally enabled, CDP packets are filtered (discarded) on CDP-disabled ports.

In the flooding mode, VLAN filtering rules are not applied, but STP rules are applied. In case of MSTP, the CDP packets are classified to instance 0.

### Example

```
console(conf) cdp run
console(conf) cdp pdu flooding
```

---

## 45.4 cdp advertise-v2

The **cdp advertise-v2** Global Configuration mode command specifies version 2 of transmitted CDP packets. The **no** format of this command specifies version 1.

### Syntax

**cdp advertise-v2**

**no cdp advertise-v2**

**Parameters**

N/A

**Default Configuration**

Version 2.

**Command Mode**

Global Configuration mode

**Example**

```
console(conf) cdp run
console(conf) cdp advertise-v2
```

---

**45.5 cdp appliance-tlv enable**

The **cdp appliance-tlv enable** Global Configuration mode command enables sending of the Appliance TLV. The **no** format of this command disables the sending of the Appliance TLV.

**Syntax****cdp appliance-tlv enable****no cdp appliance-tlv enable****Parameters**

N/A

**Default Configuration**

Enabled

**Command Mode**

Global Configuration mode

**User Guidelines**

This MIB specifies the Voice Vlan ID (VVID) to which this port belongs:

- **0** - The CDP packets transmitting through this port would contain Appliance VLAN-ID TLV with value of 0. VoIP and related packets are expected to be sent and received with VLAN-id=0 and an 802.1p priority.
- **1..4094** - The CDP packets transmitting through this port would contain Appliance VLAN-ID TLV with N. VoIP and related packets are expected to be sent and received with VLAN-ID=N and an 802.1p priority.
- **4095** - The CDP packets transmitting through this port would contain Appliance VLAN-ID TLV with value of 4095. VoIP and related packets are expected to be sent and received untagged without an 802.1p priority.
- **4096** - The CDP packets transmitting through this port would not include Appliance VLAN-ID TLV; or, if the VVID is not supported on the port, this MIB object will not be configurable and will return 4096.

### Example

```
console(conf) cdp appliance-tlv enable
```

## 45.6 cdp mandatory-tlvs validation

Use the **cdp mandatory-tlvs validation** Global Configuration mode command to validate that all mandatory (according to CDP protocol) TLVs are present in received CDP frames. The **no** format of this command disables the validation.

If the mandatory TLVs are not included in the packet, it is deleted.

### Syntax

**cdp mandatory-tlvs validation**

**no cdp mandatory-tlvs validation**

### Parameters

N/A

### Default Configuration

Enabled.

### Command Mode

Global Configuration mode

---

### Example

Turn off mandatory TLV validation:

```
console(conf) no cdp mandatory-tlvs validation
```

---

## 45.7 cdp source-interface

The **cdp source-interface** Global Configuration mode command specifies the CDP source port used for source IP address selection. The **no** format of this command deletes the source interface.

### Syntax

**cdp source-interface** *interface-id*

**no cdp source-interface**

### Parameters

**interface-id**—Source port used for Source IP address selection.

### Default Configuration

No CDP source interface is specified.

### Command Mode

Global Configuration mode

### User Guidelines

Use the **cdp source-interface** command to specify an interface whose minimal IP address will be advertised in the TVL instead of the minimal IP address of the outgoing interface.

### Example

---

```
console(conf) cdp source-interface gil
```

---

## 45.8 cdp log mismatch duplex

Use the **cdp log mismatch duplex** Global and Interface Configuration mode command to enable validating that the duplex status of a port received in a CDP

packet matches the ports actual configuration. If not, a SYSLOG duplex mismatch message is generated. The **no** format of the CLI command disables the generation of the SYSLOG messages.

### Syntax

**cdp log mismatch duplex**

**no cdp log mismatch duplex**

### Parameters

N/A

### Default Configuration

The switch reports duplex mismatches from all ports.

### Command Mode

Global Configuration mode

Ethernet Interface

### Example

---

```
console(conf) interface gil
console(conf-if) cdp log mismatch duplex
```

---

## 45.9 cdp log mismatch voip

Use the **cdp log mismatch voip** Global and Interface Configuration mode command to enable validating that the VoIP status of the port received in a CDP packet matches its actual configuration. If not, a SYSLOG message is generated by CDP. The **no** format of the CLI command disables the generation of the SYSLOG messages.

### Syntax

**cdp log mismatch voip**

**no cdp log mismatch voip**

### Parameters

N/A

### Default Configuration

The switch reports voip mismatches from all ports.

### Command Mode

Global Configuration mode

Ethernet Interface

### Example

---

```
console(conf) interface gil
console(conf-if) cdp log mismatch voip
```

---

## 45.10 cdp log mismatch native

Use the **cdp log mismatch native** Global and Interface Configuration mode command to enable validating that the native VLAN received in a CDP packet matches the actual native VLAN of the port. If not, a SYSLOG native mismatch message is generated. The **no** format of the CLI command disables the generation of the SYSLOG messages.

### Syntax

**cdp log mismatch native**

**no cdp log mismatch native**

### Parameters

N/A

### Default Configuration

The switch reports Native VLAN mismatches from all ports.

### Command Mode

Global Configuration mode, Ethernet Interface

### Example

---

```
console(conf) interface gil
```



```
console(conf-if) cdp log mismatch native
```

---

## 45.11 cdp device-id format

The **cdp device-id format** Global Configuration mode command specifies the format of the Device-ID TLV. The **no** format of this command returns to default.

### Syntax

```
cdp device-id format {mac | serial-number}
```

```
no cdp device-id format
```

### Parameters

**mac**—Specifies that the Device-ID TLV contains the device's MAC address.

**serial-number**—Specifies that Device-ID TLV contains the device's hardware serial number.

### Default Configuration

MAC address is selected by default.

### Command Mode

Global Configuration mode

### Example

---

```
console(conf) cdp device-id format serial-number
```

---

## 45.12 cdp timer

The **cdp timer** Global Configuration mode command specifies how often CDP packets are transmitted. The **no** format of this command returns to default.

### Syntax

```
cdp timer seconds
```

```
no cdp timer
```

**Parameters**

**seconds**—Value of the Transmission Timer in seconds. Range: 5-254 seconds.

**Default Configuration**

60 seconds.

**Command Mode**

Global Configuration mode

**Example**

```
console(conf) cdp timer 100
```

---

**45.13 cdp holdtime**

The **cdp holdtime** Global Configuration mode command specified a value of the Time-to-Live field into sent CDP messages. The **no** format of this command returns to default.

**Syntax**

**cdp holdtime** *seconds*

**no cdp timer**

**Parameters**

**seconds**—Value of the Time-to-Live field in seconds. The value should be bigger than the value of Transmission Timer.

**Parameters range**

**seconds**— 10 - 255.

**Default Configuration**

180 seconds.

**Command Mode**

Global Configuration mode

---

**Example**

```
console(conf) cdp holdtime 100
```

---

**45.14 clear cdp counters**

The **clear cdp counters** Global Configuration mode command resets the CDP traffic counters to 0.

**Syntax**

```
clear cdp counters
```

**Parameters**

N/A

**Command Mode**

Global Configuration mode

**Example**

---

```
console(conf) clear cdp counters
```

---

**45.15 clear cdp table**

The **clear cdp table** Global Configuration mode command deletes the CDP Cache tables.

**Syntax**

```
clear cdp table
```

**Parameters**

N/A

**Command Mode**

Global Configuration mode

---

## Example

```
console(conf) clear cdp table
```

---

## 45.16 show cdp

The **show cdp** Privileged EXEC mode command displays the interval between advertisements, the number of seconds the advertisements are valid and version of the advertisements.

### Syntax

**show cdp**

### Parameters

N/A

### Command Mode

Privileged EXEC mode

### Example

```
switch>show cdp
```

Global CDP information:

```
cdp is globally enabled
```

```
cdp log duplex mismatch is globally enabled
```

```
cdp log voice VLAN mismatch is globally enabled
```

```
cdp log native VLAN mismatch is globally disabled
```

Mandatory TLVs are

```
Device-ID TLV (0x0001)
```

```
Address TLV (0x0002)
```

```
Port-ID TLV (0x0003)
```

```
Capabilities TLV (0x0004)
```

```
Version TLV (0x0005)
```

```
Platform TLV (0x0006)
```

```
Sending CDPv2 advertisements is enabled
```

```
Sending Appliance TLV is enabled
Device ID format is Serial Number
Sending CDP packets every 60 seconds
Sending a holdtime value of 180 seconds
```

---

## 45.17 show cdp entry

The **show cdp entry** Privileged EXEC mode command displays information about specific neighbor. Display can be limited to protocol or version information

### Syntax

```
show cdp entry [* | device-name] [protocol | version]
```

### Parameters

**\***—Specifies all neighbors

**device-name**—Specifies the name of the neighbor.

**protocol**—Limits the display to information about the protocols enabled on neighbors.

**version**—Limits the display to information about the version of software running on the neighbors.

### Default Configuration

Version

### Command Mode

Privileged EXEC mode

### Example

---

#### Example 1 - .

```
switch#show cdp entry device.cisco.com
Device ID: device.cisco.com
Advertisement version: 2
Entry address(es):
```

```
IP address: 192.168.68.18
CLNS address: 490001.1111.1111.1111.00
DECnet address: 10.1
Platform: cisco 4500, Capabilities: Router
Interface: gil, Port ID (outgoing port): Ethernet0
Holdtime: 125 sec
```

**Version:**

```
Cisco Internetwork Operating System Software
IOS (tm) 4500 Software (C4500-J-M), Version 11.1(10.4), MAINTENANCE INTERIM
SOFTWARE
Copyright (c) 1986-1997 by cisco Systems, Inc.
Compiled Mon 07-Apr-97 19:51 by dschwart
```

**Example 2 -**

---

```
switch#show cdp entry device.cisco.com protocol
Protocol information for device.cisco.com:
  IP address: 192.168.68.18
  CLNS address: 490001.1111.1111.1111.00
  DECnet address: 10.1
```

---

**Example 3 -**

```
switch#show cdp entry device.cisco.com version
Version information for device.cisco.com:
  Cisco Internetwork Operating System Software
  IOS (tm) 4500 Software (C4500-J-M), Version 11.1(10.4), MAINTENANCE INTERIM
  SOFTWARE
  Copyright (c) 1986-1997 by cisco Systems, Inc.
  Compiled Mon 07-Apr-97 19:51 by dschwart
```

---

## 45.18 show cdp interface

The **show cdp interface** Privileged EXEC mode command displays information about ports on which CDP is enabled.

### Syntax

**show cdp interface** *interface-id*

### Parameters

**interface-id**—Port ID.

### Command Mode

Privileged EXEC mode

### Example

---

```
switch#show cdp interface gil
CDP is globally enabled
CDP log duplex mismatch
  Globally is enabled
  Per interface is enabled
CDP log voice VLAN mismatch
  Globally is enabled
  Per interface is enabled
CDP log native VLAN mismatch
  Globally is disabled
  Per interface is enabled
gil is Down, CDP is enabled
Sending CDP packets every 60 seconds
Holdtime is 180 seconds
```

## 45.19 show cdp neighbors

The **show cdp neighbors** Privileged EXEC mode command displays information about neighbors kept in the main or secondary cache.

### Syntax

```
show cdp neighbors [interface-id] [detail | secondary]
```

### Parameters

**interface-id**—Displays the neighbors attached to this port.

**detail**—Displays detailed information about a neighbor (or neighbors) from the main cache including network address, enabled protocols, hold time, and software version.

**secondary**—Displays information about neighbors from the secondary cache.

### Default Configuration

If interface-id is not specified, the command displays information for neighbors of all ports.

If detail or secondary are not specified, the default is secondary.

### Command Mode

Privileged EXEC mode

### Example

```
switch#show cdp neighbors
```

```
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  P - VoIP Phone, M - Remotely-Managed Device,
                  C - CAST Phone Port, W - Two-Port MAC Relay
```

Device ID	Local Interface	Adver ver	TimeToLive	Capability	Platform	Port ID
lab-7206	fa 1	2	157	R	206VXR	fa 0
lab-as5300-1	fa 1	2	163	R	AS5300	fa 0



lab-as5300-2	fa 1	2	159	R	AS5300	gi0
lab-as5300-3	fa 2	2	122	R	AS5300	gi0
lab-as5300-4	fa 2	1	132	R	AS5300	fa 0
lab-3621	fa 3	2	140	R S	31-TELKO	fa 0
008024 2758E	fa 3	2	132	T	CAT3000	fa 2

---

### switch#show cdp neighbors detail

```

-----
Device ID: lab-7206
Advertisement version: 2
Entry address(es):
    IP address: 172.19.169.83
Platform: cisco 7206VXR, Capabilities: Router
Interface: Ethernet0, Port ID (outgoing port): fa 0
Time To Live : 123 sec

Version :
Cisco Internetwork Operating System Software
IOS (tm) 5800 Software (C5800-P4-M), Version 12.1(2)
Copyright (c) 1986-2002 by Cisco Systems, Inc.
Duplex: half

-----
Device ID: lab-as5300-1
Entry address(es):
    IP address: 172.19.169.87
Platform: cisco AS5300, Capabilities: Router
--More--
.

```

---

**Gateway#show cdp neighbors fa 1 detail**

```

Device ID: SEP000427D400ED
Advertisement version: 2
Entry address(es):
  IP address: 1.6.1.81
Platform: Cisco IP Phone 7940, Capabilities: Host
Interface: fa 1, Port ID (outgoing port): Port 1
Time To Live: 150 sec
Version :
P00303020204

Duplex: full

Power drawn: 6.300 Watts

```

---

**#switch#show cdp neighbors secondary**

```

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
S - Switch, H - Host, I - IGMP, r - Repeater,
P - VoIP Phone, M - Remotely-Managed Device,
C - CAST Phone Port, W - Two-Port MAC Relay

```

Local Interface	Mac Address	TimeToLive	Capability	VLAN-ID	Platform
fa 1	00:00:01:23a:86:9c	157	R,S	10	206VXRYC
fa 1	00:00:05:53a:86:9c	163	R,S	10	ABCD-VSD
fa 3	00:00:01:23b:86:9c	140	R		QACSZ
fa 3	00:00:ab:c2a:86:9c	132	T		CAT3000

**Field Definitions:**

- **Advertisement version**—The version of CDP being used for CDP advertisements.
- **Capabilities**—The device type of the neighbor. This device can be a router, a bridge, a transparent bridge, a source-routing bridge, a switch, a host, an IGMP device, or a repeater.
- **COS for Untrusted Ports**—The COS value with which all packets received on an untrusted port should be marked by a simple switching device which cannot itself classify individual packets.
- **Device ID**—The name of the neighbor device and either the MAC address or the serial number of this device.
- **Duplex**—The duplex state of connection between the current device and the neighbor device.
- **Entry address(es)**—A list of network addresses of neighbor devices.
- **Extended Trust**—The Extended Trust.
- **External Port-ID**—Identifies the physical connector port on which the CDP packet is transmitted. It is used in devices, such as those with optical ports, in which signals from multiple hardware interfaces are multiplexed through a single physical port. It contains the name of the external physical port through which the multiplexed signal is transmitted.
- **Interface**—The protocol and port number of the port on the current device.
- **IP Network Prefix**—It is used by On Demand Routing (ODR). When transmitted by a hub router, it is a default route (an IP address). When transmitted by a stub router, it is a list of network prefixes of stub networks to which the sending stub router can forward IP packets.
- **Management Address**—When present, it contains a list of all the addresses at which the device will accept SNMP messages, including those it will only accept when received on interface(s) other than the one over which the CDP packet is being sent.
- **MTU**—The MTU of the interface via which the CDP packet is sent.
- **Native VLAN**—The ID number of the VLAN on the neighbor device.
- **Physical Location**—A character string indicating the physical location of a connector which is on, or physically connected to, the interface over which the CDP packet containing this TLV is sent.

- **Platform**—The product name and number of the neighbor device. In the case of the Secondary Cache only the 8 last characters of the value are printed.
- **Power Available**—Every switch interface transmits information in the Power Available TLV, which permits a device which needs power to negotiate and select an appropriate power setting. The Power Available TLV includes four fields.
- **Power Consumption**—The maximum amount of power, in milliwatts, expected to be obtained and consumed from the interface over which the CDP packet is sent.
- **Power Drawn**—The maximum requested power.
- **Note For IP Phones** the value shown is the maximum requested power (6.3 Watts). This value can be different than the actual power supplied by the routing device (generally 5 watts; shown using the show power command).
- **Protocol-Hello**—Specifies that a particular protocol has asked CDP to piggyback its "hello" messages within transmitted CDP packets.
- **Remote Port\_ID**—Identifies the port the CDP packet is sent on
- **sysName**—An ASCII string containing the same value as the sending device's sysName MIB object.
- **sysObjectID**—The OBJECT-IDENTIFIER value of the sending device's sysObjectID MIB object.
- **Time To Live**—The remaining amount of time, in seconds, the current device will hold the CDP advertisement from a transmitting router before discarding it.
- **Version**—The software version running on the neighbor device.
- **Voice VLAN-ID**—The Voice VLAN-ID.
- **VTP Management Domain**—A string that is the name of the collective group of VLANs associated with the neighbor device.

---

## 45.20 show cdp tlv

The **show cdp tlv** Privileged EXEC mode command displays information about TLVs sent by CDP on all ports or on a specific port.

### Syntax

```
show cdp tlv [interface-id]
```

### Parameters

**interface-id**— Port ID.

### Default Configuration

TLVs for all ports.

### Command Mode

Privileged EXEC mode

### User Guidelines

You can use the **show cdp tlv** command to verify the TLVs configured to be sent in CDP packets. The **show cdp tlv** command displays information for a single port if specified or for all ports if not specified. Information for a port is displayed if only CDP is really running on the port, i.e. CDP is enabled globally and on the port, which is UP.

### Examples:

**Example 1** - In this example, CDP is disabled and no information is displayed.

---

```
switch#show cdp tlv
cdp globally is disabled
```

---

**Example 2** - In this example, CDP is globally enabled but disabled on the port and no information is displayed.

---

```
switch#show cdp tlv gi2
cdp globally is enabled
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  P - VoIP Phone, M - Remotely-Managed Device,
                  C - CAST Phone Port, W - Two-Port MAC Relay
```

---

```
Interface TLV: gi2
CDP is disabled on gi2
```

---

**Example 3** - In this example, CDP is globally enabled and enabled on the port, but the port is down and no information is displayed.

---

```
switch#show cdp tlv interface gi2

cdp globally is enabled

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  P - VoIP Phone, M - Remotely-Managed Device,
                  C - CAST Phone Port, W - Two-Port MAC Relay

Interface TLV: gi3

CDP is enabled on gi3

Ethernet gi3 is down
```

---

**Example 4** - In this example, CDP is globally enabled and enabled on the port, which is up and information is displayed.

---

```
switch#show cdp tlv interface gi1

cdp globally is enabled

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  P - VoIP Phone, M - Remotely-Managed Device,
                  C - CAST Phone Port, W - Two-Port MAC Relay

Interface TLV: gi1

CDP is enabled

Ethernet gi1 is up,

Device ID TLV: type is MAC address; Value is 00:11:22:22:33:33:44:44

Address TLV: IPv4: 1.2.2.2 IPv6:

Port_ID TLV: gi1

Capabilities: S, I
```

```

Version TLV: 1 and 2
Platform TLV: VSD Ardd
Native VLAN TLV: 1
Full/Half Duplex TLV: full-duplex
Appliance VLAN_ID TLV: Appliance-ID is 1; VLAN-ID is 100
COS for Untrusted Ports TLV: 1
Power Available TLV: Request-ID is 1 Power management-ID is 1;
                    Available-Power is 10;
                    Management-Power-Level is 0xFFFFFFFF

```

**Example 5** - In this example, CDP is globally enabled, and no ports are specified, so information is displayed for all ports on which CDP is enabled who are up.

```

switch#show cdp tlv interface
cdp globally is enabled
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater,
                  P - VoIP Phone, M - Remotely-Managed Device,
                  C - CAST Phone Port, W - Two-Port MAC Relay

Interface TLV: gil
CDP is enabled
Ethernet gil is up,
Device ID TLV: type is MAC address; Value is 00:11:22:22:33:33:44:44
Address TLV: IPv4: 1.2.2.2 IPv6:
Port_ID TLV: gisl
Capabilities: S, I
Version TLV: 1 and 2
Platform TLV: VSD Ardd
Native VLAN TLV: 1
Full/Half Duplex TLV: full-duplex
Appliance VLAN_ID TLV: Appliance-ID is 1; VLAN-ID is 100
COS for Untrusted Ports TLV: 1

```

```
Power Available TLV: Request-ID is 1 Power management-ID is 1;
                        Available-Power is 10;
                        Management-Power-Level is 0xFFFFFFFF

Interface TLV: gi2
CDP is disabled on gi2

Interface TLV: gi3
CDP is enabled on gi3
Ethernet gi3 is down
```

---

## 45.21 show cdp traffic

The **show cdp traffic** Privileged EXEC mode command displays the CDP counters, including the number of packets sent and received and checksum errors.

### Syntax

**show cdp traffic**

### Parameters

N/A

### Command Mode

Privileged EXEC mode

### Example

---

```
switch#show cdp traffic

CDP counters:

  Total packets output: 81684,  Input: 81790

  Hdr syntax:  0, Chksum error: 0, Encaps: 0

  No memory:  0, Invalid packet: 0

  CDP version 1 advertisements output:  100,    Input  0

  CDP version 2 advertisements output:  81784,  Input  0
```



### Field Definition

- **Total packets output**—The number of CDP advertisements sent by the local device. Note that this value is the sum of the CDP Version 1 advertisements output and CDP Version 2 advertisements output fields.
- **Input**—The number of CDP advertisements received by the local device. Note that this value is the sum of the CDP Version 1 advertisements input and CDP Version 2 advertisements input fields.
- **Hdr syntax**—The number of CDP advertisements with bad headers, received by the local device.
- **Chksum error**—The number of times the checksum (verifying) operation failed on incoming CDP advertisements.
- **No memory**—The number of times the local device did not have enough memory to store the CDP advertisements in the advertisement cache table when the device was attempting to assemble advertisement packets for transmission and parse them when receiving them.
- **Invalid**—The number of invalid CDP advertisements received .
- **CDP version 1 advertisements output**—The number of CDP Version 1 advertisements sent by the local device.
- **CDP version 1 advertisements Input**—The number of CDP Version 1 advertisements received by the local device.
- **CDP version 2 advertisements output**—The number of CDP Version 2 advertisements sent by the local device.
- **CDP version 2 advertisements Input**—The number of CDP Version 2 advertisements received by the local device.

# 46 Revision History

**Table 1: Revision History Cisco 300 v.1.1**

Revision	Date	Comments
1.0	June 14, 2011	Draft



Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)