

PowerEdge VRTX 10Gb Switch Module — R1-2210 CLI Reference Guide

Notes, Cautions, and Warnings



NOTE: A NOTE indicates important information that helps you make better use of your computer.



CAUTION: A CAUTION indicates potential damage to hardware or loss of data if instructions are not followed. indicates either potential damage to hardware, or loss of data and tells you how to avoid the problem.



WARNING: A WARNING indicates a potential for property damage, personal injury, or death.

© Copyright © 2014 Dell Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. Dell and the Dell logo are trademarks of Dell Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

Contents

1	Preface	33
2	User Interface Commands	47
	enable	47
	disable	47
	login	48
	configure	49
	exit (Configuration)	49
	exit (EXEC)	50
	end	51
	help	51
	history	52
	history size	53
	terminal history	54
	terminal history size	55
	terminal datadump	56
	terminal width	57
	terminal prompt	58
	debug-mode	58
	show history	59
	show privilege	60

do	61
banner exec	62
banner login	64
banner motd	65
exec-banner	67
login-banner	68
motd-banner	69
show banner	70
3 Macro Commands	73
macro name	73
macro	76
macro description	79
macro global	80
macro global description	82
show parser macro	83
4 System Management Commands	87
ping	87
traceroute	90
telnet	93
resume	97
hostname	97
reload	98
show reload	100
service cpu-utilization	101

show cpu utilization	102
clear cpu counters	103
service cpu-counters	103
show cpu counters	104
show users	105
show sessions	106
show system	107
show version	108
show version md5	108
show system tcam utilization	109
show system defaults	109
show tech-support	110
show system sensors	111
asset-tag	112
show system id	113
5 Clock Commands	115
clock set	115
clock source	116
clock timezone	117
clock summer-time	117
sntp authentication-key	119
sntp authenticate	120
sntp trusted-key	121
sntp client poll timer	122
sntp broadcast client enable	123

sntp anycast client enable	123
sntp client enable	124
sntp client enable (Interface)	125
sntp unicast client enable	126
sntp unicast client poll	127
sntp server	127
sntp source-interface	129
sntp source-interface-ipv6	130
sntp port	131
show clock	131
show sntp configuration	133
show sntp status	134

6 Configuration & Image File Commands . 137

copy	137
write	141
delete	141
dir	142
more	143
rename	144
boot system	145
show running-config	146
show startup-config	147
show bootvar	148

7	Auto-Update and Auto-Configuration . .	151
	boot host auto-config	151
	boot host auto-update	152
	boot host dhcp	152
	boot host auto-save	153
	show boot	154
	show ip dhcp tftp-server	156
8	Management ACL Commands	159
	management access-list	159
	permit (Management)	160
	deny (Management)	162
	management access-class	163
	show management access-list	164
	show management access-class	164
9	SNMP Commands	167
	snmp-server server	167
	snmp-server community	167
	snmp-server community-group	169
	snmp-server source-interface	171
	snmp-server source-interface-ipv6	172
	snmp-server view	173
	show snmp views	175
	snmp-server group	176

show snmp groups	178
snmp-server user	179
show snmp users	181
snmp-server filter	183
show snmp filters	184
snmp-server host	185
snmp-server engineID local	187
snmp-server engineID remote	189
show snmp engineID	190
snmp-server enable traps	190
snmp-server trap authentication	191
snmp-server contact	192
snmp-server location	193
snmp-server set	193
snmp trap link-status	194
show snmp	195

10 RSA and Certificate Commands..... 199

crypto key generate dsa	200
crypto key generate rsa	201
crypto key import	202
show crypto key	204
crypto certificate generate	205
crypto certificate request	207
crypto certificate import	209
show crypto certificate	213

11 Web Server Commands	215
ip http server	215
ip http port	215
ip http timeout-policy	216
ip http secure-server	217
ip http secure-port	218
ip https certificate	218
show ip http	219
show ip https	220
ssl version	221
show ssl version	221

**12 Telnet, Secure Shell and Secure Login Commands
223**

ip telnet server	223
ip ssh server.	224
ip ssh port.	225
ip ssh password-auth.	225
ip ssh pubkey-auth	226
crypto key pubkey-chain ssh	228
user-key	229
key-string	230
show ip ssh	232
show crypto key pubkey-chain ssh.	233

13 Line Commands	235
line	235
speed	235
autobaud	236
exec-timeout	237
show line	238
14 Authentication, Authorization and Accounting	241
aaa authentication login	241
aaa authentication enable	242
login authentication	244
enable authentication	245
ip http authentication	247
show authentication methods	248
password	249
enable password	250
service password-recovery	251
username	253
show users accounts	254
aaa accounting login	255
aaa accounting dot1x	257
show accounting	259
passwords min-length	260
passwords strength-check enable	261
passwords strength minimum character-classes	262
passwords strength max-limit repeated-characters	262

passwords aging	263
passwords history	264
passwords history hold-time	265
passwords lockout	266
aaa login-history file	267
set username active	268
set line active	268
set enable-password active	269
show passwords configuration	270
show users login-history	271
15 RADIUS Commands	273
radius-server host	273
radius-server key	275
radius-server retransmit	276
radius-server host source-interface	276
radius-server host source-interface-ipv6	277
radius-server timeout	278
radius-server deadtime	279
show radius-servers	280
show radius-servers key	281
16 TACACS+ Commands	283
tacacs-server host	283
tacacs-server key	284
tacacs-server timeout	285

tacacs-server host source-interface	286
tacacs-server host source-interface-ipv6	287
show tacacs	288
show tacacs key	289
17 SYSLOG Commands	291
logging on	291
logging host	292
logging console	293
logging buffered	294
clear logging	295
logging file	295
clear logging file	296
aaa logging	297
file-system logging	298
management logging	299
show logging	299
show logging file	301
show syslog-servers	302
18 Remote Network Monitoring (RMON) Commands	305
show rmon statistics	305
rmon collection stats	307
show rmon collection stats	308
show rmon history	309

rmon alarm	312
show rmon alarm-table	314
show rmon alarm	315
rmon event	317
show rmon events	318
show rmon log	319
rmon table-size	320
19 802.1X Commands	323
aaa authentication dot1x	323
clear dot1x statistics	324
dot1x auth-not-req	325
dot1x authentication	326
dot1x guest-vlan	326
dot1x guest-vlan enable	327
dot1x guest-vlan timeout	329
dot1x host-mode	330
dot1x max-hosts	332
dot1x max-req	333
dot1x port-control	334
dot1x radius-attributes filter-id	336
dot1x radius-attributes vlan	336
dot1x re-authenticate	339
dot1x reauthentication	340
dot1x system-auth-control	340
dot1x timeout quiet-period	341

dot1x timeout reauth-period	342
dot1x timeout server-timeout	343
dot1x timeout supp-timeout	344
dot1x timeout tx-period	345
dot1x traps authentication failure	346
dot1x traps authentication success	347
dot1x violation-mode	348
show dot1x	349
show dot1x statistics	353
show dot1x users	354

20 Ethernet Configuration Commands 357

interface	357
interface range	358
shutdown	358
description	360
speed	361
duplex	362
negotiation	363
flowcontrol	364
mdix	365
back-pressure	366
port jumbo-frame	366
clear counters	367
set interface active	368
show interfaces configuration	368

show interfaces status	369
show interfaces advertise	370
show interfaces description	372
show interfaces counters	373
show ports jumbo-frame	376
storm-control broadcast enable	376
storm-control broadcast level kbps	377
storm-control include-multicast	378
show storm-control	379
21 PHY Diagnostics Commands	381
test cable-diagnostics tdr	381
show cable-diagnostics tdr	382
show cable-diagnostics cable-length	383
show fiber-ports optical-transceiver	384
22 EEE Commands	385
eee enable (global)	385
eee enable (interface)	386
eee lldp enable	386
show eee	387
23 Green Ethernet	395
green-ethernet energy-detect (global)	395
green-ethernet energy-detect (interface)	395

green-ethernet short-reach (global)	396
green-ethernet short-reach (interface)	397
green-ethernet short-reach force	398
green-ethernet short-reach threshold	399
green-ethernet power-meter reset	400
show green-ethernet	400
24 Port Channel Commands	403
channel-group	403
port-channel load-balance	404
show interfaces port-channel	405
25 Address Table Commands	407
bridge multicast filtering	407
bridge multicast mode	408
bridge multicast address	410
bridge multicast forbidden address	411
bridge multicast ip-address	412
bridge multicast forbidden ip-address	414
bridge multicast source group	415
bridge multicast forbidden source group	416
bridge multicast ipv6 mode	418
bridge multicast ipv6 ip-address	420
bridge multicast ipv6 forbidden ip-address	421
bridge multicast ipv6 source group	422
bridge multicast ipv6 forbidden source group	424

bridge multicast unregistered	425
bridge multicast forward-all	426
bridge multicast forbidden forward-all	427
mac address-table static	428
clear mac address-table	431
mac address-table aging-time	432
port security	432
port security mode	434
port security max	435
port security routed secure-address	436
show mac address-table	437
show mac address-table count	439
show bridge multicast mode	440
show bridge multicast address-table	440
show bridge multicast address-table static	444
show bridge multicast filtering	446
show bridge multicast unregistered	447
show ports security	448
show ports security addresses	449
26 Port Monitor Commands	451
port monitor	451
show ports monitor	453
27 sFlow Commands	455
sflow receiver	455

sflow flow-sampling	456
sflow counters-sampling	456
clear sflow statistics	457
show sflow configuration	458
show sflow statistics	459
28 LLDP Commands	461
lldp run	461
lldp transmit	461
lldp receive	462
lldp timer	463
lldp hold-multiplier	464
lldp reinit	465
lldp tx-delay	465
lldp optional-tlv	466
lldp optional-tlv 802.1	467
lldp management-address	469
lldp notifications	470
lldp notifications interval	471
lldp med	471
lldp med notifications topology-change	472
lldp med fast-start repeat-count	473
lldp med network-policy (global)	474
lldp med network-policy (interface)	475
clear lldp table	476
lldp med location	477

lldp chassis-id	478
show lldp configuration	479
show lldp med configuration	481
show lldp local tlvs-overloading	483
show lldp local	484
show lldp neighbors	486
show lldp statistics	491
29 Spanning-Tree Commands	495
spanning-tree	495
spanning-tree mode	495
spanning-tree forward-time	496
spanning-tree hello-time	497
spanning-tree max-age	498
spanning-tree priority	499
spanning-tree disable	500
spanning-tree cost	501
spanning-tree port-priority	502
spanning-tree portfast	502
spanning-tree link-type	503
spanning-tree pathcost method	504
spanning-tree bpdu (Global)	505
spanning-tree bpdu (Interface)	506
spanning-tree guard root	507
spanning-tree bpduguard	508
clear spanning-tree detected-protocols	509

spanning-tree mst priority	510
spanning-tree mst max-hops	511
spanning-tree mst port-priority	511
spanning-tree mst cost	512
spanning-tree mst configuration	513
instance (MST)	514
name (MST)	515
revision (MST)	516
show (MST)	516
exit (MST)	517
abort (MST)	518
show spanning-tree	519
show spanning-tree bpdu	534

30 VLAN Commands **537**

vlan database	537
vlan	538
show vlan	539
interface vlan	541
interface range vlan	541
name	542
switchport protected-port	543
show interfaces protected-ports	544
switchport community	545
switchport	546
switchport mode	547

switchport access vlan	548
switchport trunk allowed vlan	549
switchport trunk native vlan	550
switchport general allowed vlan	551
switchport general pvid	553
switchport general ingress-filtering disable	554
switchport general acceptable-frame-type	555
switchport customer vlan	556
switchport general forbidden vlan	557
map protocol protocols-group	558
switchport general map protocols-group vlan	559
show vlan protocols-groups	560
map subnet subnets-group	561
switchport general map subnets-group vlan	562
show vlan subnets-groups	563
show interfaces switchport	564
private-vlan	565
private-vlan association	566
switchport private-vlan mapping	568
switchport private-vlan host-association	569
show vlan private-vlan	570
switchport access multicast-tv vlan	571
switchport customer multicast-tv vlan	572
show vlan multicast-tv	573
ip internal-usage-vlan	574
show vlan internal usage	575

31 Voice VLAN Commands	577
voice vlan state	577
voice vlan id	578
voice vlan oui-table	579
voice vlan cos mode	580
voice vlan cos	581
voice vlan aging-timeout	582
voice vlan enable	583
voice vlan secure	584
show voice vlan	584
32 IGMP Snooping Commands	587
ip igmp snooping (Global)	587
ip igmp snooping vlan	587
ip igmp snooping vlan mrouter	588
ip igmp snooping vlan mrouter interface	589
ip igmp snooping vlan forbidden mrouter	590
ip igmp snooping vlan static	591
ip igmp snooping vlan multicast-tv	592
ip igmp snooping querier	593
ip igmp snooping querier address	594
ip igmp snooping vlan querier	595
ip igmp snooping vlan querier address	596
ip igmp snooping vlan querier election	596
ip igmp snooping vlan querier version	598
ip igmp snooping vlan immediate-leave	599

show ip igmp snooping groups	599
show ip igmp snooping interface	600
show ip igmp snooping mrouter	602
show ip igmp snooping multicast-tv	602
33 IGMP Commands	605
ip igmp last-member-query-count	605
ip igmp last-member-query-interval	606
ip igmp query-interval	607
ip igmp query-max-response-time	608
ip igmp robustness	609
34 Link Aggregation Control Protocol Commands	611
lACP system-priority	611
lACP port-priority	611
lACP timeout	612
show lACP	613
show lACP port-channel	615
35 GARP VLAN Registration Protocol Commands	617
gvrp enable (Global)	617
gvrp enable (Interface)	618
garp timer	618
gvrp vlan-creation-forbid	620
gvrp registration-forbid	620

clear gvrp statistics	621
show gvrp configuration	622
show gvrp statistics	623
show gvrp error-statistics	624
36 DHCP Snooping Commands	627
ip dhcp snooping	627
ip dhcp snooping vlan	628
ip dhcp snooping trust	628
ip dhcp snooping information option allowed-untrusted	629
ip dhcp snooping verify	630
ip dhcp snooping database	631
ip dhcp snooping database update-freq	632
ip dhcp snooping binding	632
clear ip dhcp snooping database	634
show ip dhcp snooping	634
show ip dhcp snooping binding	635
ip source-guard binding	636
ip arp inspection	637
ip arp inspection vlan	638
ip arp inspection trust	639
ip arp inspection validate	640
ip arp inspection list create	641
ip mac	642
ip arp inspection list assign	643
ip arp inspection logging interval	643

	show ip arp inspection	644
	show ip arp inspection list	645
	show ip arp inspection statistics	646
	clear ip arp inspection statistics	647
37	DHCP Relay Commands	649
	ip dhcp relay enable (Global)	649
	ip dhcp relay enable (Interface)	649
	ip dhcp relay address (Global)	650
	ip dhcp relay address (Interface)	651
	show ip dhcp relay	652
	ip dhcp information option	654
	show ip dhcp information option	655
38	DHCPv6 Commands	657
	ipv6 dhcp client stateless	657
	clear ipv6 dhcp client	658
	ipv6 dhcp client information refresh	659
	ipv6 dhcp client information refresh minimum	660
	ipv6 dhcp duid-en	661
	show ipv6 dhcp	663
	show ipv6 dhcp interface	664
39	IP Addressing Commands	667
	ip address	667

ip address dhcp	669
renew dhcp	670
ip default-gateway	671
show ip interface	672
arp	673
arp timeout (Global)	674
arp timeout(Interface)	675
clear arp-cache	676
show arp	676
show arp configuration	677
ip helper-address	678
show ip helper-address	679
show ip dhcp client interface	680
40 IPv6 Commands	683
clear ipv6 neighbors	683
ipv6 address	683
ipv6 address anycast	684
ipv6 address autoconfig	686
ipv6 address eui-64	687
ipv6 address link-local	689
ipv6 default-gateway	690
ipv6 enable	691
ipv6 hop-limit	692
ipv6 icmp error-interval	693
ipv6 link-local default zone	694

ipv6 nd advertisement-interval	695
ipv6 nd dad attempts	696
ipv6 nd hop-limit	698
ipv6 nd managed-config-flag	699
ipv6 nd ns-interval	700
ipv6 nd other-config-flag	701
ipv6 nd prefix	702
ipv6 nd ra interval	706
ipv6 nd ra lifetime	707
ipv6 nd ra suppress	708
ipv6 nd reachable-time	710
ipv6 nd router-preference	711
ipv6 neighbor	712
ipv6 redirects	714
ipv6 route	715
ipv6 unicast-routing	717
ipv6 unreachable	717
show ipv6 interface	718
show ipv6 link-local default zone	728
show ipv6 nd prefix	728
show ipv6 neighbors	730
show ipv6 route	732
show ipv6 route summary	734
show ipv6 static	735

41 IPv6 Tunnel Commands	739
interface tunnel	739
tunnel isatap solicitation-interval	739
tunnel isatap robustness	740
tunnel isatap router	741
tunnel mode ipv6ip	742
tunnel source	744
show ipv6 tunnel	745
42 IP Routing Protocol-Independent Commands	747
ip route	747
ip routing	749
show ip route	750
show ip route summary	752
43 ACL Commands	755
ip access-list (IP extended)	755
permit (IP)	756
deny (IP)	759
ipv6 access-list (IPv6 extended)	763
permit (IPv6)	764
deny (IPv6)	767
mac access-list	771
permit (MAC)	772
deny (MAC)	773

service-acl input	775
service-acl input block	776
service-acl output	778
time-range	779
absolute	781
periodic	782
show time-range	783
show access-lists	784
show interfaces access-lists	785
clear access-lists counters	785
show interfaces access-lists counters	786
.....	787
44 Quality of Service (QoS) Commands ..	789
qos	789
show qos	790
class-map	791
show class-map	792
match	793
policy-map	794
class	795
show policy-map	796
trust	797
set	799
police	800
service-policy	801

qos aggregate-policer	802
show qos aggregate-policer	804
police aggregate	804
wrr-queue cos-map	805
wrr-queue bandwidth	807
priority-queue out num-of-queues	808
traffic-shape	809
traffic-shape queue	810
rate-limit (Ethernet)	811
qos wrr-queue wrtd	812
show qos wrr-queue wrtd	813
show qos interface	813
qos wrr-queue threshold	818
qos map policed-dscp	819
qos map dscp-queue	820
qos map dscp-dp	820
qos trust (Global)	821
qos trust (Interface)	823
qos cos	823
qos dscp-mutation	824
qos map dscp-mutation	825
show qos map	826
clear qos statistics	827
qos statistics policer	828
qos statistics aggregate-policer	828
qos statistics queues	829

show qos statistics	830
45 DNS Client Commands	833
clear host	833
ip domain lookup	834
ip domain name	834
ip domain polling-interval	836
ip domain retry	837
ip domain timeout	837
ip host	838
ip name-server	840
show hosts	841
46 UDL D Commands	843
show uddl	843
uddl	847
uddl message timeout	848
uddl port	849
47 MLD Snooping Commands	851
ipv6 mld snooping (Global)	851
ipv6 mld snooping vlan	851
ipv6 mld snooping querier	852
ipv6 mld snooping vlan querier	853
ipv6 mld snooping vlan querier election	854

ipv6 mld snooping vlan querier version	855
ipv6 mld snooping vlan mrouter (Global)	856
ipv6 mld snooping vlan mrouter interface	857
ipv6 mld snooping vlan forbidden mrouter	858
ipv6 mld snooping vlan static	859
ipv6 mld snooping vlan immediate-leave	860
show ipv6 mld snooping groups	861
show ipv6 mld snooping interface	862
show ipv6 mld snooping mrouter	864
48 MLD Commands.....	865
ipv6 mld last-member-query-count	865
ipv6 mld last-member-query-interval	866
ipv6 mld query-interval	867
ipv6 mld query-max-response-time	868
ipv6 mld robustness	869
A Port Features.....	871

Preface

This guide describes how to use the CLI, and in addition, provides a list of the CLI commands and their arguments.

The CLI commands described in this document are organized according to features in separate sections.

This section describes how to use the CLI. It contains the following topics:

- User (Privilege) Levels
- CLI Command Modes
- Starting the CLI
- CLI Command Conventions
- Port Naming Convention
- Entering Commands
- IPv6 Address Conventions
- IP Address and Out-Of-Band Port

User (Privilege) Levels

Users can 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 or commands in the *Privileged EXEC* mode.
- **Level 15** — Users with this level can run all commands.

A system administrator (user with level 15) can create passwords that enables the user to go from level 1 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}
```

Using these passwords, you can raise your user level by entering the command: `enable` and the password for 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.

Examples

Create passwords for level 15 (by the administrator):

```
console#configure
console<conf># enable password level 15
level15@abc
console<conf>#
```

Create a user with user level 1:

```
console#configure
console<conf> username john password john1234
privilege 1
console<conf>
```

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#
```

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

CLI Command Modes

To configure devices, the CLI is divided into various command modes. Each command mode has its own set of specific commands. Entering a question mark "?" at the console prompt displays a list of commands available for that particular command mode.

A specific command, which varies from mode to mode, is used to navigate from one mode to another. The standard order to access the modes is as follows: *User EXEC* mode, *Privileged EXEC* mode, *Global Configuration* mode, and *Interface Configuration* modes.

When starting a session, the initial mode for non-privileged users is the *User EXEC* mode. Only a limited subset of commands is available in the *User EXEC* mode. This level is reserved for tasks that do not change the configuration.

Privileged users enter the *Privileged EXEC* mode directly using a password. This mode provides access to the device Configuration modes.

The modes are described below.

User EXEC Mode

After logging into the device, the user is automatically in *User EXEC* command mode unless the user is defined as a privileged user. In general, the *User EXEC* commands enable the user to perform basic tests, and display system information.

The user-level prompt consists of the device "host name" followed by the angle bracket (>).

```
console>
```

The default host name is "console" unless it has been changed using the `hostname` command in the *Global Configuration* mode.

Privileged EXEC Mode

Privileged access is password-protected to prevent unauthorized use, because many of the privileged commands set operating system parameters: The password is not displayed on the screen and is case sensitive.

Privileged users enter directly into the *Privileged EXEC* mode.

Use `disable` to return to the *User EXEC* mode.

Global Configuration Mode

Global Configuration mode commands apply to features that affect the system as a whole, rather than just a specific interface.

To enter the *Global Configuration* mode, enter **configure** in the *Privileged EXEC* mode, and press <Enter>.

The *Global Configuration* mode prompt is displayed.

```
console(config)#
```

Use **exit**, **end** or **ctrl/z** to return to the *Privileged EXEC* mode.

Interface Configuration Modes

Commands in the following modes perform specific interface operations:

- **Line Interface** — Contains commands to configure the management connections. These include commands such as line speed, timeout settings, etc. The *Global Configuration* mode command **line** is used to enter the *Line Configuration command* mode.
- **VLAN Database** — Contains commands to create a VLAN as a whole. The *Global Configuration* mode command **vlan database** is used to enter the *VLAN Database Interface Configuration* mode.
- **Management Access List** — Contains commands to define management access-lists. The *Global Configuration* mode command **management access-list** is used to enter the *Management Access List Configuration* mode.
- **Port Channel** — Contains commands to configure port-channels, for example, assigning ports to a VLAN or port-channel. The *Global Configuration* mode command **interface port-channel** is used to enter the *Port Channel Interface Configuration* mode.
- **SSH Public Key-Chain** — Contains commands to manually specify other device SSH public keys. The *Global Configuration* mode command **crypto key pubkey-chain ssh** is used to enter the *SSH Public Key-chain Configuration* mode.
- **Interface** — Contains commands that configure the interface. The *Global Configuration* mode command **interface** is used to enter the *Interface Configuration* mode.

Starting the CLI

The switch can be managed over a connection to the switch console port through the CMC, or via a Telnet connection. The switch is managed by entering command keywords and parameters at the prompt. Using the switch CLI commands is similar to entering commands on a UNIX system.

If access is via a Telnet connection, ensure the device has an IP address defined, corresponding management access is granted, and the workstation used to access the device is connected to the device prior to using CLI commands.

Accessing the CLI from the Console Line

- 1 Start the device and wait until the startup procedure is complete. The *User Exec* mode is entered, and the prompt `console>` is displayed.
- 2 Configure the device and enter the necessary commands to complete the required tasks.
- 3 When finished, exit the session with the `quit` or `exit` command.

Accessing the CLI from Telnet

- 1 Enter `telnet` and the IP address of the device. A User Name prompt is displayed.
- 2 Enter the User Name and Password. You are in the *Privileged Exec* mode.
- 3 Acquire the IP address of the OOB port by:

Through CLI

```
console(config-oob)#ip address dhcp
```

```
console(config-oob)#02-Oct-2006 05:05:13
%BOOTP_DHCP_CL-W-DHCPIPCCANDIDATE: The device is
waiting for IP address verification on interface
oob, IP 10.5.225.41, mask 255.255.255.224, DHCP
server 10.5.224.25
```

```
02-Oct-2006 05:05:25 %BOOTP_DHCP_CL-I-
DHCPCONFIGURED: The device has been configured on
interface oob, IP 10.5.225.41, mask
255.255.255.224, DHCP server 10.5.224.25
```

```
console (config-oob)#
```

Through CMC GUI

- a** Connect to chassis GUI.
- b** Go to I/O Module Overview -> A Gigabit Ethernet -> Setup page.
- c** Update the OOB IP parameters.
- 4** Configure the device and enter the necessary commands to complete the required tasks.
- 5** When finished, exit the session with the **quit** or **exit** command.

Accessing the CLI from CMC

- 1** From the CMC console, enter

```
$ connect switch
connect: acquiring remote port.
Connected to remote port.
Escape character is '^\''.
console#
```
- 2** Configure the device and enter the necessary commands to complete the required tasks.
- 3** To return to the CMC, enter **ctrl/**.

When another user is required to log onto the system, the **login** command is entered in the *Privileged EXEC* command mode,. This effectively logs off the current user and logs on the new user.

CLI Command Conventions

The following table describes the command syntax conventions.

Conventions	Description
[]	In a command line, square brackets indicates an optional entry.
{ }	In a command line, curly brackets indicate a selection of compulsory parameters separated by the / character. One option must be selected. For example: flowcontrol {auto on off} means that for the flowcontrol command either auto , on or off must be selected.
<i>Italic font</i>	Indicates a parameter.
<Enter>	Any individual key on the keyboard. For example click <Enter>.
Ctrl+F4	Any combination keys pressed simultaneously on the keyboard.
Screen Display	Indicates system messages and prompts appearing 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 all when no parameters are defined. For example, the command interface range port-channel 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 .
interface-id	This indicates a port, VLAN or LAG. The syntax for interface_id is as follows: <i>{port_type}port-number {vlan} vlan-id {port-channel} LAG-number</i>

Port Naming Convention

The following describes the types of ports and port naming conventions of the ports on the R1-2210 device.

Types of Ports on the R1-2210 Switch

The following ports are found on the R1-2210 switch:

- **20 x 10G Ethernet Ports.** These consist of:
 - **4 external ports**—Connected to network (visible when the switch is in the chassis)
 - **16 internal ports**—Connected to blade servers (not visible when the switch is in the chassis)
 - **2 x 1 Gigabit/s Ethernet Ports.**
- **Single Internal Out-of-Band port**

The switch supports an Out-of-Band (OOB) port that is connected to the management network of the chassis.

Port Naming Convention

There are 5 groups of 10G ports, numbered 0-4. Group 0 represents the external ports and groups 1-4 represents the internal ports that are connected to blade servers 1-4.

There are 2 1G ports.

External/Internal Ports

The following naming convention is used for internal and external ports:

tengigabitethernet group/port_number or te group/port_number

The following table maps the hardware network port numbers to the software interface port numbers and describe how they are referred to in the CLI/GUI:

Port Type and Number	Software Port Naming Convention in CLI/WEB
External 10G ports 1-4 and External 1G ports 1-2	tengigabitethernet 0/1.... tengigabitethernet 0/4. gigabitethernet 0/1, gigabitethernet 0/2
Internal 10G ports 1-4	tengigabitethernet 1/1.... tengigabitethernet 1/4
Internal 10G ports 5-8	tengigabitethernet 2/1.... tengigabitethernet 2/4

Port Type and Number	Software Port Naming Convention in CLI/WEB
Internal 10G ports 9-12	tengigabitethernet 3/1.... tengigabitethernet 3/4
Internal 10G ports 13-16	tengigabitethernet 4/1.... tengigabitethernet 4/4

Specifying Multiple Interfaces

Interface Range

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

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

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

```
console#configure
console(config-if)#interface range gi0/1-4
```

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.

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
console(config)#interface range gi0/1-4, vlan 1-2
```

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 gi0/4" show, interfaces and status are keywords, gi is an argument that specifies the interface type, and 1/0/5 is an argument that 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 smith
```

Help information can be displayed in the following ways:

- **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** — A command is incomplete and the character ? is entered in place of a parameter. The matched parameters for this command are displayed.

The following describes features that assist in using the CLI:

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. The keys that can be used to access the history buffer are described in [Keyboard Keys](#).

By default, the history buffer system is enabled, but it can be disabled at any time. For information about the command syntax to enable or disable the history buffer, see 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 256. By configuring 0, the effect is the same as disabling the history buffer system. For information about the command syntax for configuring the command history buffer, see the [history size](#) command.

To display the history buffer, see [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 guide describes the negation effect for all applicable commands.

Command Completion

If the command entered is incomplete, invalid, or has missing or invalid parameters, an appropriate error message is displayed.

To complete an incomplete command, press the <Tab> button. 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.

Incorrect or incomplete commands are automatically re-entered next to the cursor. If a parameter must be added, the parameter can be added to the basic command already displayed next to the cursor. The following example indicates that the command interface requires a missing parameter.

```
(config)#interface
%missing mandatory parameter
(config)#interface
```

Keyboard Shortcuts

The CLI has a range of keyboard shortcuts to assist in entering the CLI commands.

The following table describes these shortcuts:

Keyboard Key	Description
Up-arrow key	Recalls commands from the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Down-arrow key	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 <i>Privileged EXEC</i> mode from any mode.
Backspace key	Moves the cursor back one space.
Up-arrow key	Recalls commands from the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.

IPv6 Address Conventions

The following describes how to write an IPv6z address, which is a link-local IPv6 address:

The format is: `<ipv6-link-local-address>%<egress-interface>`

where:

egress-interface (also known as zone) = `vlan<vlan-id> | po <number> | tunnel <number> | port<number> | 0`

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

The following combinations are possible:

- `ipv6_address%egress-interface`—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.

IP Address and Out-Of-Band Port

The switch supports an Out-of-Band (OOB) port. This port is used for the management network.

The IP address assigned to this port cannot be assigned to the in-band ports at the same time. In addition, the OOB port has a default IP address assigned to it: 192.168.2.1 /24. This default IP address is used when no other address was assigned (dynamically or statically). This subnet is a reserved one and cannot be assigned on the in-band interfaces.

User Interface Commands

enable

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

Syntax

enable

Parameters

N/A

Default Configuration

The default privilege level is 15.

Command Mode

User EXEC mode

Example

The following example enters privilege level 15.

```
console# enable  
enter password:*****  
console# Accepted
```

disable

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

Syntax

disable

Parameters

N/A

Default Configuration

The default privilege level is 15.

Command Mode

Privileged EXEC mode

Example

The following example returns the user to user level 1.

```
console# disable 1
```

login

The **login** User 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

User EXEC mode

Example

The following example enters Privileged EXEC mode and logs in with the required username 'bob'.

```
console# login
User Name:bob
Password:*****
console#
```

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
```

exit (Configuration)

The `exit` command exits any mode and brings the user to the next higher mode in the CLI mode hierarchy.

Syntax

```
exit
```

Parameters

N/A

Default Configuration

N/A

Command Mode

All 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
```

exit (EXEC)

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

Syntax

exit

Parameters

N/A

Default Configuration

N/A

Command Mode

User EXEC mode

Example

The following example closes an active terminal session.

```
console# exit
```

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
```

help

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

Syntax

help

Parameters

N/A

Default Configuration

N/A

Command Mode

All configuration 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?').
-

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.

It is effective from the next time that the user logs in via console/telnet/ssh.

The following are related commands:

- Use the [terminal history size](#) User 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 size of the command history buffer.

Example

The following example enables the command for Telnet.

```
console(config)# line telnet  
console(config-line)# history
```

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.

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. It is effective from the next time that the user logs in via console/telnet/ssh.

Use the [terminal history size](#) User 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
```

terminal history

The **terminal history** User 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

User 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.

This command is effective immediately.

Example

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

```
console# terminal no history
```

terminal history size

The **terminal history size** User 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

User 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
```

terminal datadump

The **terminal datadump** User 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

terminal no datadump

Parameters

N/A

Default Configuration

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

Command Mode

User 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 not limited, 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
```

terminal width

Use the **terminal width** User EXEC mode command to determine the width of the display for the echo input to CLI sessions. Use **terminal no width** to return to the default.

The command is per session and will not be saved in the configuration database.

Syntax

terminal width *number-of-characters*

terminal no width

Parameters

number-of-characters - Specifies the number of characters to be displayed for the echo output of the CLI commands and the configuration file, '0' means endless number of characters on a screen line. (Range: 0, 70-512)

Default Configuration

The default number of characters is 77.

Command Mode

Privileged EXEC mode

Example

The following example sets the terminal width to 100 characters

```
console# terminal width 100
```

terminal prompt

Use the **terminal prompt** User EXEC mode command to enable the terminal prompts. Use **terminal no prompt** command to disable the terminal prompts.

The command is per session and will not be saved in the configuration database.

Syntax

terminal prompt

terminal no prompt

Parameters

N/A

Default Configuration

The default configuration is prompts enabled.

Command Mode

Privileged EXEC mode

Example

The following example disables the terminal prompts

```
console# terminal no prompt
```

debug-mode

The **debug-mode** Privileged EXEC mode command mode switches to debug mode.

Syntax

debug-mode

Parameters

N/A

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example enters Debug mode.

```
console# debug-mode
```

show history

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

Syntax

show history

Parameters

N/A

Default Configuration

N/A

Command Mode

User 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)
```

show privilege

The `show privilege` User EXEC mode command displays the current privilege level.

Syntax

```
show privilege
```

Parameters

N/A

Default Configuration

N/A

Command Mode

User EXEC mode

Example

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

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

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	Authorizat ion
----	----	-----	----	-----
1	1	te0/1-4,Po1,Po2	other	Required
2	2	te0/1	dynamicGvrp	Required
10	v0010	te0/1	permanent	Not Required
11	V0011	te0/1,te0/3	permanent	Required

20	20	te0/1	permanent	Required
30	30	te0/1,te0/3	permanent	Required
31	31	te0/1	permanent	Required
91	91	te0/1,te0/4	permanent	Required
4093	guest- vlan	te0/1,te0/3	permanent	Guest

```
console(config)#
```

banner exec

Use the **banner exec** Global Configuration mode command to specify and enable a message to be displayed after a successful logon. This banner is applied automatically on all the user interfaces: console, Telnet and SSH and also on the WEB GUI. Use the **no** form of this command to delete the existing EXEC banner.

Syntax

```
banner exec d message-text d
```

```
no banner exec
```

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**—The message must start in 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, press **<Enter>** to continue).

Default Configuration

Disabled (no EXEC 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 exec** Line Configuration command to disable the Exec banner on a particular line or lines.

Example

The following example sets an EXEC 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.

```
console(config)# banner exec %
```

```
Enter TEXT message. End with the character '%'.  
$(bold)Session activated. $(bold) Enter commands at the prompt.  
%
```

```
$(bold)Session activated. $(bold) Enter commands at the prompt.  
%
```

When a user logs on to the system, the following output is displayed:

```
Session activated. Enter commands at the prompt.
```

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 user 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.

```
console(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
```

banner motd

Use the **banner motd** command in Global Configuration mode to specify and enable a message-of-the-day banner. This message is displayed before the

login banner. Use the **no** form of this command to delete the existing MOTD banner.

Syntax

banner motd *d message-text d*

no banner motd

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**—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. Tokens are described in the User Guidelines. The message can contain up to 2000 characters (after every 510 characters, you must press <Enter> to continue).

Default Configuration

Disabled (no MOTD 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.

When a user connects to a device, the message-of-the-day (MOTD) banner appears first, followed by the login banner and prompts. After the user logs in to the device, the EXEC banner is displayed.

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.

Token	Information displayed in the banner
\$(bold)	Indicates that the next text is a bold text. Using this token again to 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 motd** Line Configuration command to disable the MOTD banner on a particular line or lines.

Example

The following example sets an MOTD 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.

```
console(config)# banner motd %
```

```
Enter TEXT message. End with the character '%'.
$(bold)Upgrade$(bold) to all devices begins at March 12
%
```

```
$(bold)Upgrade$(bold) to all devices begins at March 12
```

```
%
```

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

```
Upgrade to all devices begins at March 12
```

exec-banner

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

Syntax

```
exec-banner
```

```
no exec-banner
```

Parameters

N/A

Default Configuration

Disabled

Command Mode

Line Configuration Mode

Example

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

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
```

motd-banner

Use the **motd-banner** command in Line Configuration Mode to enable the display of message-of-the-day banners. Use the **no** form of this command to disable the display of MOTD banners.

Syntax

motd-banner

no motd-banner

Parameters

N/A

Default Configuration

Enabled

Command Mode

Line Configuration Mode

Example

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

show banner

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

Syntax

```
show banner motd
show banner login
show banner exec
```

Parameters

N/A

Command Mode

User EXEC mode

Examples

```
console# show banner motd
Banner: MOTD
Line SSH: Enabled
Line Telnet: Enabled
Line Console: Enabled
```

10000 giga ports switch

```
console# show banner login
```

Banner: Login

Line SSH: Enabled

Line Telnet: Enabled

Line Console: Enabled

```
console# show banner exec
```

Banner: EXEC

Line SSH: Enabled

Line Telnet: Enabled

Line Console: Enabled

You have logged on

Macro Commands

macro name

Use the **macro name** Global Configuration mode command to define a macro. If a macro with this name already exists, it overrides the previously-defined one.

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

Global macros define a group of CLI commands that can be run at any time

Syntax

macro name *macro-name*

no macro name [*macro-name*]

Parameters

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

Default Configuration

N/A

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.

Keywords

Macros may contain keywords (parameters). The following describes keywords:

- A macro can contain up to three keywords.
- All matching occurrences of the keyword are replaced by the corresponding value specified in the **macro** command.

- Keyword matching is case-sensitive
- Applying a macro with keywords does not change the state of the original macro definition.

User Feedback

The behavior of a macro command requiring user feedback is the same as if the command is entered from terminal: it sends its prompt to the terminal and accepts the user reply.

Creating a Macro

Use the following guidelines to create a macro:

- Use **macro name** to create the macro with the specified name.
- Enter one macro command per line.
- Use the @ character to end the macro.
- Use the # character at the beginning of a line to enter a comment in 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/description pairs. The keywords and descriptions are displayed in the GUI pages when the macro is displayed.
- The syntax for this preprocessor command is as follows:
- **#macro key description** *\$keyword1 description1 \$keyword2 description2 \$keyword3 description3*
- A keyword must be prefixed with '\$'.
- **#macro keywords** - This instruction enables the device to display the keywords as part of the CLI help. It 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** and **macro global** 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 $keyword1 $keyword2 $keyword3
```

where \$keywordn is the name of the keyword.

Editing a Macro

Macros cannot be edited. Modify a macro by creating a new macro with the same name as the existing macro. The newer macro overwrites the existing macro.

Scope of Macro

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 a few exceptions, there are other ways of executing macros in the various configuration modes. Macros may be executed in Privileged Exec mode, Global Configuration mode, and Interface Configuration mode (when the interface is NOT a VLAN.)

Examples

Example 1—The following example shows how to create a macro that configures the duplex mode of a port.

```
console(config)# macro name dup
Enter macro commands one per line. End with the character '@'.
#macro description dup
duplex full
negotiation
@
```

Example 2—The following example shows how to create a macro with 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.

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

```
no negotiation
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) and then run the macro on the port. The **#macro keywords** command entered in the macro definition enables the user to receive help for the macro, as shown after the words e.g. below.

```
console(config)# interface te0/1
console(config-if)# macro apply duplex ?
WORD <1-32> Keyword to replace with value e.g. $DUPLEX, $SPEED
<cr>
console(config-if)# macro apply duplex $DUPLEX ?
WORD<1-32> First parameter value
<cr>
console(config-if)# macro apply duplex $DUPLEX full $SPEED ?
WORD<1-32> Second parameter value
console(config-if)# macro apply duplex $DUPLEX full $SPEED 100
```

macro

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*—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 (Ethernet, Port Channel) Configuration mode

User Guidelines

The **macro apply** command hides the commands of the macro from the user while it is being run. The **macro trace** command displays the commands along with any errors which are generated by them as they are executed. This is used to debug the macro and find syntax or configuration errors.

When you run a macro, if a line in it fails 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. The **show parser macro** command displays the macro history of an interface.

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 attempted to be applied and may fail or succeed on the remaining interfaces.

Examples.

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

```
console(config)# interface te0/2
console(config-if)# macro trace dup $DUPLEX full $SPEED 100
  Applying command... 'duplex full'
  Applying command... 'speed 100'
console(config-if)#
```

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

```
console(config)# interface te0/2
console(config-if)# macro apply dup $DUPLEX full $SPEED 100
console(config-if)#
```

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

```
console(config)# interface te0/1
console(config-if)# macro trace dup
Applying command...'duplex full'
Applying command...'speed auto'
% bad parameter value
console(config-if)#
```

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 (Ethernet, Port Channel) 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.

To verify the settings created by this command, run the **show parser macro** command.

Example

```
console(config)# interface te0/2
console(config-if)# macro apply dup
console(config-if)# exit
console(config)# interface te0/3
```

```

console(config-if)# macro apply duplex $DUPLEX full $SPEED 100
console(config-if)# macro description dup
console(config-if)# macro description duplex
console(config-if)# end
console(config)# exit
console# show parser macro description
Global Macro(s):
Interface      Macro Description(s)
-----
te0/2          dup
te0/3          duplex | dup | duplex
-----

console# configure
console(config)# interface te0/2
console(config-if)# no macro description
console(config-if)# end
console(config)# exit
console# show parser macro description
Global Macro(s):
Interface      Macro Description(s)
-----
te0/3          duplex | dup | duplex
-----

```

macro global

Use the **macro global** Global Configuration command to apply a macro to a switch (with or without the trace option).

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*—Specify the parameter values required for 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

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.

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 this command 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. Use **show parser macro** to display the global macro history.

Example.

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

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

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

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

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 / detailed] / 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.
- **detailed**—Displays information for non-present ports in addition to present ports.

Default Configuration

Display description of all macros on present ports.

If the **detailed** keyword is not used, only present ports are displayed.

Command Mode

User EXEC mode

Examples

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

```
console# show parser macro
```

```
Total number of macros = 6
```

```
-----  
Macro name : company-global
```

```
Macro type : default global
```

```
# Enable dynamic port error recovery for link state  
# failures
```

```
-----  
Macro name : company-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
```

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

```
console# 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.

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

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

```
console# show parser macro description
Global Macro(s): company-global
```

Example 5 - This is an example of output from the `show parser macro description interface` command.

```
console# show parser macro description interface te0/2
Interface Macro Description
-----
te0/2 this is test macro
-----
```


System Management Commands

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] [source source-address]
```

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

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. See [IPv6 Address Conventions](#).
- **hostname**—Hostname to ping (Length: 1-160 characters. Maximum label size for each part of the host name: 58.)
- **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).
- **source *source-address***—Source address (Unicast IPv4 address or global Unicast IPv6 address).

Default Usage

N/A

Command Mode

Privileged 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.

See [IPv6 Address Conventions](#).

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.

When the `source` keyword is configured and the source address is not an address of the switch, the command is halted with an error message and pings are not sent.

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 66.218.71.198: icmp_seq=0. time=11 ms  
64 bytes from 66.218.71.198: icmp_seq=1. time=8 ms  
64 bytes from 66.218.71.198: icmp_seq=2. time=8 ms  
64 bytes from 66.218.71.198: 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 FF02::1: icmp_seq=1. time=0 ms  
64 bytes from FF02::1: icmp_seq=1. time=70 ms  
64 bytes from FF02::1: icmp_seq=2. time=0 ms  
64 bytes from FF02::1: icmp_seq=1. time=1050 ms  
64 bytes from FF02::1: icmp_seq=2. time=70 ms
```

```
64 bytes from FF02::1: icmp_seq=2. time=1050 ms
64 bytes from FF02::1: icmp_seq=3. time=0 ms
64 bytes from FF02::1: icmp_seq=3. time=70 ms
64 bytes from FF02::1: icmp_seq=4. time=0 ms
64 bytes from FF02::1: icmp_seq=3. time=1050 ms
64 bytes from FF02::1: icmp_seq=4. time=70 ms
64 bytes from FF02::1: icmp_sq=4. time=1050 ms
---- FF02::1 PING Statistics----
4 packets transmitted, 12 packets received
```

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.
- **ipv6-address**—IPv6 address of the destination host.
- **hostname**—Hostname of the destination host. (Length: 1-160 characters. Maximum label size for each part of the host name: 58.)
- **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

Privileged 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

```
console# 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  kscopyng-snvang.abilene.ucaid.edu (198.32.8.103) 33 msec 35 msec
35 msec
 5  iplsng-kscopyng.abilene.ucaid.edu (198.32.8.80) 47 msec 45 msec
45 msec
 6  so-0-2-0x1.aa1.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.

telnet

The **telnet** EXEC mode command logs 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 size for each part of the host name: 58.)
- **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.

Command Mode

Privileged 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
/echo	Enables local echo.
/quiet	Prevents onscreen display of all messages from the software.
/source-interface	Specifies the source interface.
/stream	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.
Ctrl-shift-6 x	Returns to the System Command Prompt.

Ports Table

Keyword	Description	Port Number
BCP	Border Gateway Protocol	179
chargen	Character generator	19
cmd	Remote commands	514
daytime	Daytime	13
discard	Discard	9
domain	Domain Name Service	53
echo	Echo	7

Keyword	Description	Port Number
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
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
```

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

Privileged EXEC mode

Example

The following command switches to open Telnet session number 1.

```
console# resume 1
```

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: 1-160 characters. Maximum label size for each part of the host name: 58). 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)#
```

reload

The **reload** Privileged EXEC mode command reloads the operating system at a user-specified time.

Syntax

reload [**in** [hhh:mm | mmm] | **at** hh:mm [day month]] | **cancel**]

Parameters

- **in** hhh:mm | mmm - Schedules a reload of the software to take effect in the specified minutes or hours and minutes. The reload must take place within approximately 24 days.
- **at** hh:mm - Schedules a reload of the software to take place at the specified time (using a 24-hour clock). If you specify the month and day, the reload is scheduled to take place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on

the current day (if the specified time is later than the current time) or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight. The reload must take place within 24 days.

- **day** - Number of the day in the range from 1 to 31.
- **month** - Month of the year.
- **cancel** - Cancels a scheduled reload.

Default Usage

N/A

Command Mode

Privileged EXEC mode

User Guidelines

The **at** keyword can be used only if the system clock has been set on the device. To schedule reloads across several devices to occur simultaneously, synchronize the time on each device with SNTP.

When you specify the reload time using the **at** keyword, if you specify the month and day, the reload takes place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on the current day (if the specified time is later than the current time), or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight. The reload must take place within 24 days.

To display information about a scheduled reload, use the **show reload** command.

Examples

Example 1: 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) [Y]
```

Example 2: The following example reloads the operating system in 10 minutes.

```
console# reload in 10
```

```
This command will reset the whole system and disconnect your
current session. Reload is scheduled for 11:57:08 UTC Fri
Apr 21 2012 (in 10 minutes). Do you want to continue? (y/n)
[Y]
```

Example 3: The following example reloads the operating system at 13:00.

```
console# reload at 13:00
```

```
This command will reset the whole system and disconnect your
current session. Reload is scheduled for 13:00:00 UTC Fri
Apr 21 2012 (in 1 hour and 3 minutes). Do you want to
continue? (y/n) [Y]
```

Example 4: The following example cancels a reload.

```
console# reload cancel
```

```
Reload cancelled.
```

show reload

The `show reload` Privileged EXEC mode command displays whether there is a pending reload for status of the device.

Syntax

```
show reload
```

Parameters

```
N/A
```

Default Usage

```
N/A
```

Command Mode

Privileged EXEC mode

User Guidelines

You can use this command to display a pending software reload. To cancel a pending reload, use this command with the **cancel** parameter.

Example

The following example displays that reboot is scheduled for 00:00 on Saturday, April-20.

```
console# show reload
Reload scheduled for 00:00:00 UTC Sat April 20 (in 3 hours
and 12 minutes)
```

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 enabled.

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
```

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%

clear cpu counters

The `clear cpu counters` EXEC mode command clears traffic counters to and from the CPU.

Syntax

`clear cpu counters`

Parameters

N/A

Default Usage

N/A

Command Mode

User EXEC mode

Example

The following example clears the CPU traffic counters.

```
console# clear cpu counters
```

service cpu-counters

The `service cpu-counters` Global Configuration mode command enables traffic counting to and from the CPU. To disable counting, use the `no` form of this command.

Syntax

`service cpu-counters`

`no service cpu-counters`

Parameters

N/A

Default Usage

N/A

Command Mode

Global Configuration mode

User Guidelines

Use the `show cpu counters` command to display the CPU traffic counters.

Example

The following example enables counting CPU traffic.

```
console(config)# service cpu-counters
```

show cpu counters

The `show cpu counters EXEC` mode command displays traffic counter information to and from the CPU.

Syntax

`show cpu counters`

Parameters

N/A

Default Usage

N/A

Command Mode

User EXEC mode

User Guidelines

Use the `service cpu-counters` command to enable traffic counting to and from the CPU.

Example

The following example displays the CPU traffic counters.

```
console# show cpu counters
CPU counters are active.
In Octets: 987891
In Unicast Packets: 3589
In Multicast Packets: 29
In Broadcast Packets: 8
Out Octets: 972181
Out Unicast Packets: 3322
Out Multicast Packets: 22
Out Broadcast Packets: 8
```

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

User 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

show sessions

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

Syntax

`show sessions`

Parameters

N/A

Default Usage

N/A

Command Mode

User 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.

show system

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

Syntax

```
show system
```

Command Mode

User EXEC mode

Example

```
console# show system
```

```
System Description:
```

```
System Type:
```

```
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:          1.1.3.6
Unit Temperature (Celsius) Status
-----
1      42                    OK
```

show version

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

Syntax

```
show version
```

Command Mode

User 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 A01
```

show version md5

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

Syntax

```
show version md5
```

Command Mode

User EXEC mode

Example

```
console# show version md5
```

Filename	Status	MD5 Digest
image1	Active	23FA000012857D8855AABC7577AB5562
image2	Not Active	23FA000012857D8855AABEA7451265456
boot		23FA000012857D8855AABC7577AB8999

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
```

Default Usage

`unit-id`—Specifies the unit number. (Range: 1–0)

Command Mode

User EXEC mode

Example

The following example displays TCAM utilization information.

show system defaults

Use the `show system defaults` EXEC mode command to display system defaults.

Syntax

```
show system defaults [session]
```

Parameters

session—Show information for specific session only. Available values are: management, 802.lx, port, fdb, port-mirroring, spanning-tree, vlan, voice-vlan, ip-addressing, network-security and qos-acl.

Command Mode

User EXEC mode

Examples

```
console# show system defaults
```

show tech-support

Use the **show tech-support** EXEC mode command to display system and configuration information that can be provided to the Technical Assistance Center when reporting a problem.

Syntax

```
show tech-support [config | memory]
```

Parameters

- **memory**—Displays memory and processor state data.
- **config**—Displays switch configuration within the CLI commands supported on the device.

Default Configuration

By default, this command displays the output of technical-support-related show commands. Use keywords to specify the type of information to be displayed. If you do not specify any parameters, the system displays all configuration and memory data.

Command Types

Switch command.

Command Mode

User EXEC mode

User Guidelines

△ **CAUTION:** Avoid running multiple **show tech-support** commands on a switch or multiple switches on the network segment. Doing so may cause starvation of some time sensitive protocols, such as STP.

The **show tech-support** command may time out if the configuration file output takes longer to display than the configured session time out time. If this happens, enter a **set logout timeout** value of 0 to disable automatic disconnection of idle sessions or enter a longer timeout value.

The **show tech-support** command output is continuous, meaning that it does not display one screen at a time. To interrupt the output, press Esc.

If the user specifies the **memory** keyword, the **show tech-support** command displays the following output:

- Flash info (dir if exists, or flash mapping)
- Output of command **show bootvar**
- Buffers info (like **print os buff**)
- Memory info (like **print os mem**)
- Proc info (like **print OS tasks**)
- Versions of software components
- Output of command **show cpu utilization**

show system sensors

Use the **show system sensors EXEC** mode command to view the temperature sensor status

Syntax

show system sensors

Parameters

N/A

Default Usage

N/A

Command Mode

User EXEC mode

Example

For Standalone systems with a single sensor status

```
console# show system sensors
```

Temperature Sensor Type	Current Temperature (C)	Target Temperature (C)
Ambient 1	36	70
Component 1	59	95
Component 2	24	90
Component 3	30	105
Component 4	25	105

asset-tag

The **asset-tag** Global Configuration mode command assigns an asset-tag to a device. Use the **no** form of this command to restore the default setting.

Syntax

```
asset-tag tag
```

```
no asset-tag
```

Parameters

- **tag**—Specifies the device asset-tag.

Default Configuration

No asset tag is defined.

Command Mode

Global Configuration mode

Example

The following example assigns the asset-tag 2365491870 to the device.

```
console(config)# asset-tag 2365491870
```

show system id

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

Syntax

`show system id`

Command Mode

User EXEC mode

Example

The following example displays the system identity information.

```
console# show system id  
Service Tag: 89788978  
Serial number: 8936589782  
Asset tag: 7843678957
```


Clock Commands

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)

Default Configuration

The time of the image creation.

Command Mode

Privileged EXEC mode

User Guidelines

After boot the system clock is set to the time of the image creation.

If an external clock source, or an SNTP time server is not defined, the manual clock setting is not persistent across boots.

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
```

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.

If no parameter is specified, SNTP will be configured as the time source.

Command Mode

Global Configuration mode

User Guidelines

After boot the system clock is set to the time of the image creation.

Example

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

```
console(config)# clock source sntp
console(config)# exit
console# show clock
*10:46:48 UTC May 28 2013
Time source is sntp
```

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
```

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 1–5, first to 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
```

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)
- *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
console(config)# sntp trusted-key 8
console(config)# sntp authenticate
```

sntp authenticate

The `sntp authenticate` Global Configuration mode command enables authentication for received 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
console(config)# sntp authentication-key 8 md5 ClkKey
console(config)# sntp trusted-key 8
```

sntp trusted-key

The **sntp trusted-key** Global Configuration mode command authenticates the identity of the system with which 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
console(config)# sntp authentication-key 8 md5 ClkKey
console(config)# sntp trusted-key 8
console(config)# sntp authenticate
```

sntp client poll timer

The `sntp client poll timer` Global Configuration mode command sets the polling time for the SNTP client. Use the `no` form of this command to restore the default configuration.

Syntax

`sntp client poll timer seconds`

`no sntp client poll timer`

Parameters

- *seconds*—Specifies the polling interval in seconds. (Range: 60–86400).

Default Configuration

The default polling interval is 1024 seconds.

Command Mode

Global Configuration mode

Example

The following example sets the polling time for the SNTP client to 120 seconds.

```
console(config)# sntp client poll timer 120
```

sntp broadcast client enable

The `sntp broadcast client enable` Global Configuration mode command enables 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`

Parameters

N/A

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 the `clock source sntp` command for the command to be run. If this command is not run, the switch will not synchronize with Broadcast servers.

Example

The following example enables SNTP Broadcast clients.

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

sntp anycast client enable

The `sntp anycast client enable` Global Configuration mode command enables the SNTP Anycast client. Use the `no` form of this command to disable the SNTP Anycast client.

Syntax

`sntp anycast client enable`

Parameters

N/A

Default Configuration

The SNTP anycast client is disabled.

Command Mode

Global Configuration mode

User Guidelines

Use this command to enable the SNTP Anycast client.

Example

The following example enables SNTP Anycast clients.

```
console(config)# sntp anycast client enable
```

sntp client enable

The `sntp client enable` Global Configuration mode command enables the SNTP Broadcast and Anycast client. Use the `no` form of this command to disable the SNTP Broadcast and Anycast client.

Syntax

`sntp client enable interface-id`

`no sntp client enable interface-id`

Parameters

- *interface-id*—Specifies an interface ID, which can be one of the following types: Ethernet port, Port-channel or VLAN.

Default Configuration

The SNTP client is disabled on an interface.

Command Mode

Global Configuration mode

Interface Configuration mode

User Guidelines

Use the `sntp client enable` command to enable SNTP Broadcast and Anycast clients.

Example

The following example enables the SNTP Broadcast and Anycast clients on VLAN 100:

```
console(config)# sntp client enable vlan 100
```

sntp client enable (Interface)

To enable the SNTP Broadcast and Anycast client on an interface, use the `sntp client enable` Interface Configuration command. Use the `no` form of this command to disable the SNTP client.

Syntax

`sntp client enable`

`no sntp client enable`

Parameters

N/A

Default Configuration

The SNTP client is disabled on an interface.

Command Mode

Interface Configuration mode

User Guidelines

This command enables the SNTP Broadcast and Anycast client on an interface. Use the `no` form of this command to disable the SNTP client.

Example

The following example enables the SNTP broadcast and anycast client on an interface.

```
console(config)# interface vlan 100  
console(config-if)# sntp client enable  
console(config-if)# exit
```

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
```

sntp unicast client poll

The `sntp unicast client poll` Global Configuration mode command enables polling for the SNTP predefined Unicast clients. Use the `no` form of this command to disable the polling for the SNTP client.

Syntax

`sntp unicast client poll`

`no sntp unicast client poll`

Parameters

N/A

Default Configuration

Polling is disabled.

Command Mode

Global Configuration mode

User Guidelines

Polling time is configured with the `sntp client poll timer` Global Configuration mode command.

Example

The following example enables polling for SNTP predefined unicast clients.

```
console(config)# sntp unicast client poll
```

sntp server

The `sntp server` Global Configuration mode command configures the device to use the 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 {ip-address | hostname} [poll] [key keyid]`

`no sntp server [ip-address | hostname]`

Parameters

- *ip-address*—Specifies the server IP address. This can be an IPv4, IPv6 or IPv6z address.
- *hostname*—Specifies the server hostname. Only translation to IPv4 addresses is supported. (Length: 1–158 characters. Maximum label length for each part of the hostname: 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

Use the `sntp server {ip-address | hostname} [poll] [key keyid]` command to define a SNTP server. The switch supports up to 8 SNTP servers.

Use the `no sntp server ip-address | hostname` command to remove one SNTP server.

Use the `no sntp server` to remove all SNTP servers.

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
```

sntp source-interface

Use the **sntp source-interface** Global Configuration mode command to specify the source interface whose IPv4 address will be used as the source IPv4 address for communication with IPv4 SNTP servers. Use the **no** form of this command to restore the default configuration.

Syntax

sntp source-interface *interface-id*

no sntp source-interface

Parameters

- *interface-id*—Specifies the source interface.

Default Configuration

The source IPv4 address is the IPv4 address defined on the outgoing interface and belonging to next hop IPv4 subnet.

Command Mode

Global Configuration mode

User Guidelines

If the source interface is the outgoing interface, the interface IP address belonging to next hop IPv4 subnet is applied.

If the source interface is not the outgoing interface, the minimal IPv4 address defined on the interface is applied.

If there is no available IPv4 source address, a SYSLOG message is issued when attempting to communicate with an IPv4 SNTP server.

Example

The following example configures the VLAN 10 as the source interface.

```
console(config)# sntp source-interface vlan 10
```

sntp source-interface-ipv6

Use the `sntp source-interface-ipv6` Global Configuration mode command to specify the source interface whose IPv6 address will be used as the Source IPv6 address for communication with IPv6 SNTP servers. Use the `no` form of this command to restore the default configuration.

Syntax

`sntp source-interface-ipv6 interface-id`

`no sntp source-interface-ipv6`

Parameters

- *interface-id*—Specifies the source interface.

Default Configuration

The IPv6 source address is the IPv6 address defined of the outgoing interface and selected in accordance with RFC6724.

Command Mode

Global Configuration mode

User Guidelines

The outgoing interface is selected based on the SNTP server's IP address. If the source interface is the outgoing interface, the IPv6 address defined on the interfaces and selected in accordance with RFC 6724.

If the source interface is not the outgoing interface, the minimal IPv4 address defined on the interface and with the scope of the destination IPv6 address is applied.

If there is no available IPv6 source address, a SYSLOG message is issued when attempting to communicate with an IPv6 SNTP server.

Example

The following example configures the VLAN 10 as the source interface.

```
console(config)# sntp source-interface-ipv6 vlan 10
```

sntp port

The **sntp port** Global Configuration mode command specifies a SNTP User Datagram Protocol (UDP) port. Use the **no** form of this command to use the SNTP server default port.

Syntax

sntp port *port-number*

no sntp port

Parameters

- *port-number*—Specifies the UDP port number used by an SNTP server. (Range 1–65535).

Default Configuration

The default port number is 123.

Command Mode

Global Configuration mode

Example

The following example specifies that port 321 of the SNTP server is the UDP port.

```
console(config)# sntp port 321
```

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 time zone and summer time configuration.

Command Mode

User EXEC mode

User Guidelines

Before the time, there is displayed either a star (*), period (.), or blank:

- star (*)—The clock is invalid.
- period (.)—The clock was set manually.
- blank—The clock was set by SNTP.

Examples

Example 1 - 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
```

Example 2 - The following example displays the system time and date along with the time zone and summer time configuration.

```
console# show clock detail  
15:22:55 SUN Apr 23 2012  
Time source is sntp  
Time zone (DHCPv4 on VLAN1):  
Acronym is RAIN  
Offset is UTC+2  
Time zone (Static):  
Offset is UTC+0  
Summertime (DHCPv4 on VLAN1):  
Acronym is SUN  
Recurring every year.
```

Begins at first Sunday of Apr at 02:00.
Ends at first Tuesday of Sep at 02:00.
Offset is 60 minutes.
Summertime (Static):
Acronym is GMT
Recurring every year.
Begins at first Sunday of Mar at 10:00.
Ends at first Sunday of Sep at 10:00.
Offset is 60 minutes.
DHCP timezone: Enabled

show sntp configuration

The `show sntp configuration` Privileged EXEC mode command displays the SNTP configuration on the device.

Syntax

`show sntp configuration`

Parameters

N/A

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Examples

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

```
console# show sntp configuration
SNTP port : 123
Polling interval: 1024 seconds
MD5 Authentication Keys
```

```
-----  
2   John123  
3   Alice456  
-----
```

Authentication is not required for synchronization.

No trusted keys

Unicast Clients: enabled

Unicast Clients Polling: enabled

Server: 1.1.1.121

 Polling: disabled

 Encryption Key: disabled

Server: dns_server1.comapany.com

 Polling: enabled

 Encryption Key: disabled

Broadcast Clients: enabled for IPv4 and IPv6

Anycast Clients: disabled

No Broadcast Interfaces

Source IPv4 interface: vlan 1

Source IPv6 interface: vlan 10

show sntp status

The `show sntp status` Privileged EXEC mode command displays the 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: 176.1.1.8
```

```
Source: DHCPv4 on VLAN 1
```

```
Status: Up
```

```
Last response: 19:58:22.289 PDT Feb 19 2005
```

```
Stratum Level: 1
```

```
Offset: 7.33mSec
```

```
Delay: 117.79mSec
```

```
Server: dns_server.comapany.com
```

```
Source: static
```

```
Status: Unknown
```

```
Last response: 12:17:17.987 PDT Feb 19 2005
```

```
Stratum Level: 1
```

```
Offset: 8.98mSec
```

```
Delay: 189.19mSec
```

```
Anycast servers:
```

```
Server: 176.1.11.8
```

```
Interface: VLAN 112
```

```
Status: Up
```

```
Last response: 9:53:21.789 PDT Feb 19 2005
```

```
Stratum Level: 10
```

```
Offset: 9.98mSec
```

```
Delay: 289.19mSec
```

```
Broadcast servers:
```

```
Server: 3001:1::12
```

```
Interface: VLAN 101
```

```
Last response: 9:53:21.789 PDT Feb 19 2005
```

```
Stratum Level: 255
```


Configuration & Image File Commands

copy

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

Syntax

`copy source-url destination-url`

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).

The following URL options are supported:

- **running-config**—Currently running configuration file.
- **startup-config**, **flash://startup-config**—Startup configuration file.
- **image**, **flash://image**—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.
- **boot**—Boot file.
- **tftp://**—Source or destination URL for a TFTP network server. The syntax for this alias is `tftp://host/[directory]/filename`. The host can be either an IP address or a host name.
- **null:**—Null destination for copies or files. A remote file can be copied to null to determine its size. For instance `copy running-conf null` returns the size of the running configuration file.
- **xmodem:**—Source for the file from a serial connection that uses the Xmodem protocol

Default Configuration

Sensitive data is excluded if no method was specified

Command Mode

Privileged EXEC mode

User Guidelines

The location of the 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.

IPv6z Address Format

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.

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 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.

Various Copy Options Guidelines

- **Copying an Image File from a Server to Flash Memory**

Use the **copy source-url 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 Running Configuration File**

Use the **copy source-url running-config** command to load a configuration file from a network server to the running configuration file of the device. The commands in the loaded configuration file are added to those in the running configuration file as if the commands were typed in the command-line interface (CLI). The resulting configuration file is a combination of the previous running configuration and the loaded configuration files, with the loaded configuration file taking precedence.

- **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.

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]

```

write

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

Syntax

`write` [*memory*]

Parameters

This command has no arguments or keywords.

Command Mode

Privileged EXEC mode

Examples

The following example shows how to overwrite the startup-config file with the running-config file with the `write` command.

```
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
```

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)

Command Mode

Privileged EXEC mode

User Guidelines

The following keywords and URL prefixes are supported

- **flash://**—URL of the FLASH file.
- **startup-config**—Startup configuration file.
- **WORD**—Name of file (e.g. backup-config).

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

Example

The following example deletes the file called ‘backup-config’ from the flash memory.

```
console# delete flash://backup-config  
Delete flash:backup-config? [confirm]
```

dir

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

Syntax

dir

Parameters

This command has no arguments or keywords.

Command Mode

Privileged EXEC mode

Example

Example 1. The following example displays the list of files on a flash file system with static images. The Flash size column for all files except dynamic image specifies the maximum allowed size. The Data size column for dynamic images specifies the real size in the FLASH occupied by the file.

```
console# dir
Directory of flash:
File Name      Permission  Flash Size  Data Size    Modified
-----      -
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
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
```

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).

Default Configuration

This command has no arguments or keywords.

Command Mode

Privileged EXEC mode

User Guidelines

The following keywords and URL prefixes are supported

- **flash://**—URL of the FLASH file.
- **startup-config**—Startup configuration file.
- **WORD**—Name of file (e.g. backup-config).

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 te0/1-4
speed 1000
exit
no lldp run
line console
exec-timeout 0
```

rename

The **rename** Privileged EXEC mode command renames a file.

Syntax

```
rename url new-url
```

Parameters

- *url*—Specifies the file location URL. (Length: 1–160 characters)
- *new-url*—Specifies the file's new URL. (Length: 1–160 characters)

Default Configuration

This command has no arguments or keywords.

Command Mode

Privileged EXEC mode

User Guidelines

The following keywords and URL prefixes are supported

- **flash://**—URL of the FLASH file.
- **startup-config**—Startup configuration file.
- **WORD**—Name of file (e.g. backup-config).

*.sys and *.prv files cannot be renamed.

Example

The following example renames the configuration backup file.

```
console# rename backup-config m-config.bak
```

boot system

The **boot system** Privileged EXEC mode command specifies the active system image file that will be 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. The results of this command is displayed in `show bootvar`.

```
console# boot system image-1
```

show running-config

Use the `show running-config` privileged EXEC command to display the contents of the currently running configuration file.

```
show running-config
```

Parameters

This command has no arguments or keywords.

Command Mode

Privileged EXEC mode

Example

The following example displays the running configuration file contents.

```
console# show running-config
config-file-header
AA307-02
v1.2.5.76 / R750_NIK_1_2_584_002
CLI v1.0
no spanning-tree
interface range te0/1-4
```

```
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#
```

show startup-config

Use the **show startup-config** Privileged EXEC mode command to display the Startup Configuration file contents.

Syntax

```
show startup-config
```

Parameters

This command has no arguments or keywords.

Command Mode

Privileged EXEC mode

Example

The following example displays the startup configuration file contents.

```
console# show startup-config
config-file-header
AA307-02
v1.2.5.76 / R750_NIK_1_2_584_002
CLI v1.0
no spanning-tree
interface range te0/1-4
```

```
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#
```

show bootvar

Use the `show bootvar` EXEC mode command to display the active system image file that was loaded by the device at startup, and to display the system image file that will be loaded after rebooting the switch.

Syntax

```
show bootvar
```

```
show bootvar [unit unit-id]
```

Parameters

This command has no arguments or keywords.

Command Mode

User EXEC mode

Example

The following example displays the active system image file that was loaded by the device at startup and the system image file that will be loaded after rebooting the switch:

```
console# show bootvar
Image  Filename      Version      Date              Status
-----  -
```

1	image-1	1.1.0.73	19-Jun-2011	18:10:49	Not active*
2	image-2	1.1.0.73	19-Jun-2011	18:10:49	Active

"*" designates that the image was selected for the next boot

Auto-Update and Auto-Configuration

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 with the `auto` option.

Command Mode

Global Configuration mode

User Guidelines

The TFTP protocol is used to download/upload a configuration file.

Example

Example. The following example enables auto configuration via DHCP:

```
console# boot host auto-config
```

boot host auto-update

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

Syntax

boot host auto-update

no boot host auto-update

Parameters

N/A

Default Configuration

Enabled by default with the **auto** option.

Command Mode

Global Configuration mode

User Guidelines

The TFTP protocol is used to download/upload an image file.

Example

Example—The following example enables auto update via DHCP:

```
console# boot host auto-update
```

boot host dhcp

Use the **boot host dhcp** Global Configuration mode command to force downloading a configuration file at the next system startup. Use the **no** form of this command to restore the host configuration file to the default.

Syntax

boot host dhcp

no boot host dhcp

Parameters

N/A

Default Configuration

Disabled.

Command Mode

Global Configuration mode

User Guidelines

Configuring `boot host dhcp` does not take effect until the next reboot.

Example

```
console# boot host dhcp
```

boot host auto-save

Use the `boot host auto-save` Global Configuration mode command to automatically save the Running configuration file in the Startup configuration file after download. Use the `no` form of this command to restore default behavior.

Syntax

`boot host auto-save`

`no boot host auto-save`

Parameters

N/A

Default Configuration

Disable

Command Mode

Global Configuration mode

Examples

```
console# boot host auto-save
```

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

Privileged EXEC mode

Examples

```
console# show boot
Auto Config
-----
Config Download via DHCP: enabled
Next Boot Config Download via DHCP: force
Configuration file auto-save: enabled
Auto Config State: Finished successfully
Server IP address: 1.2.20.2
Configuration filename: /config/configfile1.cfg
    Auto Update
    -----
```

Image Download via DHCP: enabled

console# **show boot**

Auto Config

Config Download via DHCP: enabled

Next Boot Config Download via DHCP: force

Configuration file auto-save: enabled

Auto Config State: Opening <hostname>-config file

Auto Update

Image Download via DHCP: enabled

console# **show boot**

Auto Config

Config Download via DHCP: enabled

Next Boot Config Download via DHCP: force

Configuration file auto-save: enabled

Auto Config State: Downloading configuration file

Auto Update

Image Download via DHCP: enabled

console# **show boot**

Auto Config

Config Download via DHCP: enabled

Next Boot Config Download via DHCP: force

Configuration file auto-save: enabled

Auto Config State: Searching device hostname in indirect file

Auto Update

Image Download via DHCP: enabled

```
console# show boot
```

```
Auto Config
```

```
-----
```

```
Config Download via DHCP: enabled
```

```
Next Boot Config Download via DHCP: force
```

```
Configuration file auto-save: enabled
```

```
    Auto Update
```

```
-----
```

```
Image Download via DHCP: enabled
```

```
Auto Update State: Downloaded indirect image file
```

```
Indirect Image filename: /image/indirectimage.txt
```

show ip dhcp tftp-server

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

Syntax

```
show ip dhcp tftp-server
```

Parameters

N/A

Default Configuration

N/A

Command Mode

User EXEC mode

User Guidelines

The backup server can be a TFTP server.

Example

```
show ip dhcp tftp-server
```

```
tftp server address
```

```
active      1.1.1.1 from sname
```

```
file path on tftp server
```

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


Management ACL Commands

management access-list

The **management access-list** Global Configuration mode command configures a management access list (ACL) and enters the Management Access-list Configuration 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 **mlist**, configures management te0/1 and te0/2, and makes the new access list the active list.

```
console(config)# management access-list mlist
console(config-macl)# switchpermit te0/1
console(config-macl)# switchpermit te0/2
console(config-macl)# switchexit
console(config)# management access-class mlist
```

Example 2 - The following example creates a management access list called 'mlist', configures all interfaces to be management interfaces except te0/1 and te0/2, and makes the new access list the active list.

```
console(config)# management access-list mlist
console(config-macl)# switchdeny te0/1
console(config-macl)# switchdeny te0/2
console(config-macl)# switchpermit
console(config-macl)# switchexit
console(config)# management access-class mlist
```

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)# switchpermit
```

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 **mlist**.

```
console(config)# management access-list mlist
console(config-macl)# switchdeny
```

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 **mlist** as the active management access list.

```
console(config)# management access-class mlist
```

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 `m1` management ACL.

```
console# show management access-list m1
m1
--
deny service telnet
permit te0/1 service telnet
! (Note: all other access implicitly denied)
console(config-macl)#
```

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
```


SNMP Commands

snmp-server server

Use the `snmp-server server` Global Configuration mode command to enable the device to be configured by the SNMP protocol. 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
```

snmp-server community

Use the `snmp-server community` Global Configuration mode command to set the community access string (password) that permits access to SNMP commands (v1 and v2). This is used for SNMP commands, such as GETs and SETs.

This command configures both SNMP v1 and v2.

Use the `no` form of this command to remove the specified community string.

Syntax

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

no snmp-server community *community-string* [*ip-address*] [**type** {**router** | **oob**}]

Parameters

- **community-string**—Define the password that 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 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)
- **ip-address**—Management station IP address. The default is all IP addresses. This can be an IPv4 address, IPv6 or IPv6z address. See [IPv6 Address Conventions](#).
- **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.
- **type** *router*—Indicates whether the IP address is on the out-of-band or in-band network.

Default Configuration

No community is defined

Command Mode

Global Configuration mode

User Guidelines

The logical key of the command is the pair (community, ip-address). If ip-address is omitted, the key is (community, All-IPs). This means that there cannot be two commands with the same community, ip address pair.

The *view-name* is used to restrict the access rights of a community string. When a view-name is specified, 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),

Example

Defines a password for administrator access to the management station at IP address 1.1.1.121 and mask 255.0.0.0.

```
console(config)# snmp-server community abcd su 1.1.1.121 mask 255.0.0.0
```

snmp-server community-group

Use `snmp-server community-group` to configure access rights to a user group. The group must exist in order to be able to specify the access rights. This command configures both SNMP v1 and v2.

Syntax

```
snmp-server community-group community-string group-name [ip-address / ipv6-address] [mask mask / prefix prefix-length] [type {router | oob}]
```

Parameters

- **community-string**—Define the password that permits access to the SNMP protocol. (Range: 1–20 characters).
- **ip-address**—Management station IP address. The default is all IP addresses. This can be an IPv4 address, IPv6 or IPv6z address. See [IPv6 Address Conventions](#).
- **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**—This is the name of a group configured using [snmp-server group](#) with v1 or v2 (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***—Indicates whether the IP address is on the out-of-band or in-band network.

Default Configuration

No community is defined

Command Mode

Global Configuration mode

User Guidelines

The *group-name* is used to restrict the access rights of a community string. When a group-name is specified, the software:

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

Example

Defines a password *tom* for the group *abcd* that enables this group to access the management station 1.1.1.121 with prefix 8.

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

snmp-server source-interface

To specify the interface from which a Simple Network Management Protocol (SNMP) trap originates the informs or traps, use the **snmp-server source-interface** command in Global Configuration mode. To returned to the default, use the **no** form of this command.

Syntax

```
snmp-server source-interface {traps | informs} interface-id  
no snmp-server source-interface [traps | informs]
```

Parameters

- **traps**—Specifies the SNMP traps interface.
- **informs**—Specifies the SNMP informs.
- *interface-id*—Specifies the source interface.

Default Configuration

The source IPv4 address is the IPv4 address defined on the outgoing interface and belonging to next hop IPv4 subnet.

If no parameters are specified in **no snmp-server source-interface**, the default is both traps and informs.

Command Mode

Global Configuration mode

User Guidelines

If the source interface is the outgoing interface, the interface IP address belonging to next hop IPv4 subnet is applied.

If the source interface is not the outgoing interface, the minimal IPv4 address defined on the source interface is applied.

If there is no available IPv4 source address, a SYSLOG message is issued when attempting to send an SNMP trap or inform.

Use the **no snmp-server source-interface traps** command to remove the source interface for SNMP traps.

Use the **no snmp-server source-interface informs** command to remove the source interface for SNMP informs.

Use the **no snmp-server source-interface** command to remove the source interface for SNMP traps and informs.

Example

The following example configures the VLAN 10 as the source interface for traps.

```
console(config)# snmp-server source-interface traps vlan 100
```

snmp-server source-interface-ipv6

To specify the interface from which a Simple Network Management Protocol (SNMP) trap originates the informs or traps, use the **snmp-server source-interface** command in Global Configuration mode. To returned to the default, use the **no** form of this command.

Syntax

```
snmp-server source-interface-ipv6 {traps | informs} interface-id
```

```
no snmp-server source-interface-ipv6 [traps | informs]
```

Parameters

- **traps**—Specifies the SNMP traps interface.
- **informs**—Specifies the SNMP traps informs.
- *interface-id*—Specifies the source interface.

Default Configuration

The IPv6 source address is the IPv6 address of the outgoing interface and selected in accordance with RFC6724.

If no parameters are specified in **no snmp-server source-interface**, the default is both traps and informs.

Command Mode

Global Configuration mode

User Guidelines

If the source interface is the outgoing interface, the IPv6 address defined on the interfaces is selected in accordance with RFC 6724.

If the source interface is not the outgoing interface, the minimal IPv6 address defined on the source interface with the scope of the destination IPv6 address is applied.

If there is no available IPv6 source address, a SYSLOG message is issued when attempting to send an SNMP trap or inform.

Use the **no snmp-server source-interface-ipv6 traps** command to remove the source IPv6 interface for SNMP traps.

Use the **no snmp-server source-interface-ipv6 informs** command to remove the source IPv6 interface for SNMP informs.

Use the **no snmp-server source-interface-ipv6** command to remove the source IPv6 interface for SNMP traps and informs.

Example

The following example configures the VLAN 10 as the source interface.

```
console(config)# snmp-server source-interface-ipv6 traps vlan  
100
```

snmp-server view

The **snmp-server view** Global Configuration mode command creates or updates an SNMP view. Use the **no** form of this command to remove an SNMP view.

Syntax

`snmp-server view view-name oid-tree {included / excluded}`

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

Parameters

- **view-name**—Specifies the name for the view that is being created or updated. (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 and, optionally, a sequence of numbers. Replace a single sub-identifier with the asterisk (*) wildcard to specify a subtree family; for example 1.3.*.4. This parameter depends on the MIB being specified.
- **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.

The command's logical key is the pair (view-name, oid-tree). Therefore there cannot be two commands with the same view-name and 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 (this format is specified on the parameters specified in ifEntry).

```
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
```

show snmp views

Use the `show snmp views` Privileged EXEC mode command to display the SNMP views.

Syntax

```
show snmp views [viewname]
```

Parameters

viewname—Specifies the view name. (Length: 1–30 characters)

Default Configuration

If **viewname** is not specified, all views are displayed.

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

snmp-server group

Use the `snmp-server group` Global Configuration mode command to configure an SNMP group. Groups are used to map SNMP users to SNMP views. Use the `no` form of this command to remove an 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] }
```

Parameters

- `group 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 that no packet authentication will be performed. Applicable only to the SNMP version 3 security model.
- `auth`—Specifies that packet authentication without encryption will be performed. Applicable only to the SNMP version 3 security model.
- `priv`—Specifies that packet authentication with encryption will be performed. Applicable only to the SNMP version 3 security model. Note that creation of SNMPv3 users with both authentication and privacy must be done in the GUI. All other users may be created in the CLI.
- `notify notifyview`—Specifies the view name that enables generating informs or a traps. An inform is a trap that requires acknowledgement. Applicable only to the SNMP version 3 security model. (Length: 1–30 characters)

- **read** *readview*—Specifies the view name that enables viewing only. (Length: 1–30 characters)
- **write** *writeview*—Specifies the view name that enables configuring the agent. (Length: 1–30 characters)

Default Configuration

No group entry exists.

If *notifyview* is not specified, the notify view is not defined.

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

If *writeview* is not specified, the write view is not defined.

Command Mode

Global Configuration mode

User Guidelines

The group defined in this command is used in the [snmp-server user](#) command to map users to the group. These users are then automatically mapped to the views defined in this command.

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, assigns the encrypted security level to the group, and limits the access rights of a view called *user-view* to read-only. User *tom* is then assigned to *user-group*. So that user *tom* has the rights assigned in *user-view*.

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

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)

Default Configuration

Display all groups.

Command Mode

Privileged EXEC mode

Example

The following example displays the configured SNMP groups:

```
console# show snmp groups
```

Name	Security	Views			
	Mode	Level	Read	Write	Notify
-----	1	----	-----	-----	-----
user-	----	no_auth	Default	" "	" "
group	V2	no_auth	Default	Default	" "
managers-	V2				
group					

The following table describes significant fields shown above.

Field	Description	
Name	Group name.	
Security	Model	SNMP model in use (v1, v2 or v3).
Security	Level	Packet security. Applicable to SNMP v3 security only.

Field		Description
Views	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.

snmp-server user

Use the `snmp-server user` Global Configuration mode command to configure a new SNMP Version user. Use the `no` form of the command to remove a user. Use the `encrypted` form of this command to enter the authentication and privacy passwords in encrypted form (see SSD).

Syntax

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

```
no snmp-server user username {v1 | v2c | [remote host] v3}auth {md5 |
sha}
```

Parameters

- **username**—Define 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 `v1` or `v2c` parameters (no specific order of the 2 command configurations is imposed on the user). (Range: Up to 30 characters)
- **v1**—Specifies that the user is a v1 user.
- **v2c**—Specifies that the user is a v2c user.
- **v3**—Specifies that the user is a v3 user.
- **remote host**—IP address (IPv4, IPv6 or IPv6z) or host name of the remote SNMP host. See [IPv6 Address Conventions](#).
- **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 password. Range: Up to 32 characters.
- **priv-password**—Specifies the privacy password (The encryption algorithm used is data encryption standard - DES). Range: Up to 64 characters.

Default Configuration

No group entry exists.

Command Mode

Global Configuration mode

User Guidelines

For SNMP v1 and v2, this command performs the same actions as `snmp-server community-group`, except that `snmp-server community-group` configures both v1 and v2 at the same time. With this command, you must perform it once for v1 and once for v2.

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

A local SNMP EngineID must be defined in order to add SNMPv3 users to the device (use the `snmp-server engineID remote` command). For remote hosts users a remote SNMP EngineID is also required (use the `snmp-server engineID remote` command).

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, because an inform is a trap that requires acknowledgement. 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

This example assigns user *tom* to group *abcd* using SNMP v1 and v2c. The default is assigned as the engineID. User *tom* is assigned to group *abcd* using SNMP v1 and v2c

```
console(config)# snmp-server user tom acbd v1
console(config)# snmp-server user tom acbd v2c
console(config)# snmp-server user tom acbd v3
```

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)

Default Configuration

Display all users.

Command Mode

Privileged EXEC mode

Example

The following examples displays the configured SNMP users

```
console# show snmp users
```

```
User name                :ulrem
  Group name              :group1
  Authentication Algorithm : None
  Privacy Algorithm       : None
  Remote                  :11223344556677
  Auth Password           :
  Priv Password           :

User name                : qqg
  Group name              : www
  Authentication Algorithm : MD5
  Privacy Algorithm       : None
  Remote                  :
  Auth Password           : helloworld1234567890987665
  Priv Password           :

User name                : hello
  Group name              : world
  Authentication Algorithm : MD5
  Privacy Algorithm       : DES
  Remote                  :
  Auth Password (encrypted):
Z/tC3UF5j0pYfmXm8xeMvcIOQ6LQ4GOACCGYLRdAgOE6XQKTC
                                qMlrnpWuHraRlZj
  Priv Password (encrypted) :
kNlZHhSLo6WWxlkuZVzhL0o1gI5waaNf7Vq6yLBpJdS4N68tL
                                1tbTRSz2H4c4Q4o

User name                : ulnoAuth
  Group name              : group1
  Authentication Algorithm : None
  Privacy Algorithm       : None
  Remote                  :
  Auth Password (encrypted) :
```

```

Priv Password (encrypted)      :
User name                      : u1OnlyAuth
Group name                     : group1
Authentication Algorithm       : SHA
Privacy Algorithm              : None
Remote                         :
Auth Password (encrypted):
8nPzy2hzuba9pG3iiC/q045lRynUn7kq94L9WORFrRM=
Priv Password (encrypted) :

```

snmp-server filter

The **snmp-server filter** Global Configuration mode command creates or updates an SNMP server notification filter. Use the **no** form of this command to remove a notification filter.

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 filter in other commands. (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. 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 (this format depends on the parameters define in ifEntry).

```
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
```

show snmp filters

Use the `show snmp filters` Privileged EXEC mode command to display the defined SNMP filters.

Syntax

```
show snmp filters [filename]
```

Parameters

filename—Specifies the filter name. (Length: 1–30 characters)

Default Configuration

If *filename* is not defined, all filters are displayed.

Command Mode

Privileged EXEC mode

Example

The following example displays the configured SNMP filters.

```
console# show snmp filters user-filter
```

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

snmp-server host

Use the `snmp-server host` Global Configuration mode command to configure the host for SNMP notifications: (traps/informs). Use the `no` form of this command to remove the specified host.

Syntax

```
snmp-server host {host-ip / 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 {ip-address / hostname} [traps | informs] [version {1 |  
2c | 3}]
```

Parameters

- **host-ip**—IP address of the host (the targeted recipient). The default is all IP addresses. This can be an IPv4 address, IPv6 or IPv6z address. See [IPv6 Address Conventions](#).
- **hostname**—Hostname of the host (the targeted recipient). (Range: 1–158 characters. Maximum label size of each part of the host name: 63)
- **trap**—Sends SNMP traps to this host (default).
- **informs**—Sends SNMP informs to this host. An inform is a trap that requires acknowledgement. Not applicable to SNMPv1.
- **version 1**—SNMPv1 traps are used.
- **version 2c**—SNMPv2 traps or informs are used

- *version 3*—SNMPv2 traps or informs are used
- Authentication options are available for SNMP v3 only. The following options are available:
 - **noauth**—Specifies no authentication of a packet.
 - **auth**—Specifies authentication of a packet without encryption.
 - **priv**—Specifies authentication of a packet with encryption.
- **community-string**—Password-like community string sent with the notification operation. (Range: 1–20 characters). For v1 and v2, any community string can be entered here. For v3, the community string must match the user name defined in `snmp-server user (ISCLI)` command for v3.
- **udp-port** *port*—UDP port of the host to use. The default is 162. (Range: 1–65535)
- **filter** *filtername*—Filter for this host. If unspecified, nothing is filtered. The filter is defined using `snmp-server filter` (no specific order of commands is imposed on the user). (Range: Up to 30 characters)
- **timeout** *seconds*—(For informs only) Number of seconds to wait for an acknowledgment before resending informs. The default is 15 seconds. (Range: 1–300)
- **retries** *retries*—(For informs only) Maximum number of times to resend an inform request, when a response is not received for a generated message. The default is 3. (Range: 0–255)

Default Configuration

Version: SNMP V1

Type of notification: Traps

udp-port: 162

If informs are specified, the default for retries: 3

Timeout: 15

Command Mode

Global Configuration mode

User Guidelines

The logical key of the command is the list (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 does not automatically create a user or a notify view.

Use the commands `snmp-server user` (ISCLI) and `snmp-server group` to create a user or a group.

Example

The following defines a host at the IP address displayed.

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

snmp-server engineID local

The `snmp-server engineID local` Global Configuration mode command specifies the SNMP engineID on the local device for SNMP v3. Use the `no` form of this command to remove this engine ID.

Syntax

```
snmp-server engineID local {engineid-string | default}
```

```
no snmp-server engineID local
```

Parameters

- **engineid-string**—Specifies a concatenated hexadecimal character string identifying the engine ID. Each byte in a hexadecimal character string is two hexadecimal digits. Bytes are separated by a period or colon. If an odd number of hexadecimal digits are entered, the system automatically prefixes the digit 0 to the string. (Length: 5–32 characters, 9–64 hexadecimal digits)
- **default**—Specifies that the engine ID is created automatically based on the device MAC address.

Default Configuration

The default engine ID is defined per standard as:

- First 4 octets: First bit = 1, the rest is IANA Enterprise number = 674.
- Fifth octet: Set to 3 to indicate the MAC address that follows.
- Last 6 octets: The device MAC address.

Command Mode

Global Configuration mode

User Guidelines

To use SNMPv3, an engine ID must be specified for the device. Any ID can be specified or the default string, which is generated using the device MAC address, can be used.

As the engineID should be unique within an administrative domain, the following guidelines are recommended:

- Since the engineID should be unique within an administrative domain, use the default keyword to configure the Engine ID or configure it explicitly. In the latter case verify that it is unique within the administrative domain.
- Changing or removing the value of `snmpEngineID` deletes the SNMPv3 users database.
- The SNMP EngineID cannot be all 0x0 or all 0xF or 0x00000001.

Example

The following example enables SNMPv3 on the device and sets the device local engine ID to the default value.

```
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
```

snmp-server engineID remote

To specify the 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 ip-address engineid-string
```

```
no snmp-server engineID remote ip-address
```

Parameters

- `ip-address`—IPv4, IPv6 or IPv6z address of the remote device. See [IPv6 Address Conventions](#).
- `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 remote engineID is not configured by default.

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.

Example

```
console(config)# snmp-server engineID remote 1.1.1.1  
11:AB:01:CD:23:44
```

show snmp engineID

Use the `show snmp engineID` Privileged EXEC mode command to display the local SNMP engine ID.

Syntax

`show snmp engineID`

Parameters

N/A

Default Configuration

N/A

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
```

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 all SNMP traps.

Syntax

`snmp-server enable traps`

`no snmp-server enable traps`

Default Configuration

SNMP traps are enabled.

Command Mode

Global Configuration mode

User Guidelines

If `no snmp-server enable traps` has been entered, you can enable failure traps by using `snmp-server trap authentication` as shown in the example.

Example

The following example enables SNMP traps except for SNMP failure traps.

```
console(config)# snmp-server enable traps  
console(config)# no snmp-server trap authentication
```

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
```

Parameters

N/A

Default Configuration

SNMP failed authentication traps are enabled.

Command Mode

Global Configuration mode

User Guidelines

The command `snmp-server enable traps` enables all traps including failure traps. Therefore, if that command is enabled (it is enabled by default), this command is not necessary.

Example

The following example disables all SNMP traps and enables only failed authentication traps.

```
console(config)# no snmp-server enable traps
console(config)# snmp-server trap authentication
```

snmp-server contact

Use the `snmp-server contact` Global Configuration mode command to set the value of 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 system contact information. (Length: 1–160 characters)

Default Configuration

N/A

Command Mode

Global Configuration mode

Example

The following example sets the system contact information to `Technical_Support`.

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

snmp-server location

Use the **snmp-server location** Global Configuration mode command to set the value of 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 the system location information. (Length: 1–160 characters)

Default Configuration

N/A

Command Mode

Global Configuration mode

Example

The following example sets the device location to `New_York`.

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

snmp-server set

Use the **snmp-server set** Global Configuration mode command to define SNMP MIB commands in the configuration file if a MIB performs an action for which there is no corresponding CLI command.

Syntax

snmp-server set *variable-name name value* [*name2 value2...*]

Parameters

- **variable-name**—Specifies an SNMP MIB variable name, which must be a valid string.

- **name value**—Specifies a list of names 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.

Default Configuration

N/A

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 CLI command. To generate configuration files that support those situations, the system uses [snmp-server set](#). This command is not intended for the end user.

Example

The following example configures the scalar MIB sysName with the value TechSupp.

```
console(config)# snmp-server set sysName sysname TechSupp
```

snmp trap link-status

Use the `snmp trap link-status` Interface Configuration mode command to enable link-status generation of SNMP traps. Use the `no` form of this command to disable generation of link-status SNMP traps.

Syntax

```
snmp trap link-status
```

```
no snmp trap link-status
```

Parameters

N/A

Default Configuration

Generation of SNMP link-status traps is enabled

Command Mode

Interface Configuration mode

Example

The following example disables generation of SNMP link-status traps.

```
console(config)# interface te0/1
console(config-if)# # no snmp trap link-status
```

show snmp

Use the `show snmp` Privileged EXEC mode command to display the SNMP status.

Syntax

`show snmp`

Parameters

N/A

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example displays the SNMP communications status.

```
console# show snmp
```

```
SNMP is enabled
```

```
SNMP traps Source IPv4 interface: vlan 1
```

```
SNMP informs Source IPv4 interface: vlan 11
```

```
SNMP traps Source IPv6 interface: vlan 10
```

```
SNMP informs Source IPv6 interface:
```

Community-String	Community-Access	View name	IP Address	Mask
-----	-----	-----	-----	----
public	read only	user-view	All	
private	read write	Default	172.16.1.1/10	
private	su	DefaultSuper	172.16.1.1	

Community-string	Group name	IP Address	Mask	Type
-----	-----	-----		-----
public	user-group	All		Router

```
Traps are enabled.
```

```
Authentication trap is enabled.
```

```
Version 1,2 notifications
```

Target Address	Type	Communi-ty	Versi-on	UDP Port	Filter Name	TO Sec	Retri-es
-----	----	-----	-----	----	-----	---	-----
192.122.173.42	Trap	public	2	162		15	
192.122.173.42	Inform	public	2	162		15	3
							3

```
Version 3 notifications
```

```

Target Address      Type      Usernam  Secur  UDP      Filter  TO      Retri
-----            ----      e        ity    Port     name    Sec     es
192.122.173.42    Inform    -        -      162     -       15     -----
                  Bob      Priv    -      -      -       -       -

```

```

System Contact: Robert
System Location: Marketing

```

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

Field	Description
Community-string	The community access string permitting access to SNMP.
Community-access	The permitted access type—read-only, read-write, super access.
IP Address	The management station IP Address.
Target Address	The IP address of the targeted recipient.
Version	The SNMP version for the sent trap.

RSA and Certificate Commands

Keys and Certificates

The device automatically generates default RSA/DSA keys and certificates at following times:

- When the device is booted following a software upgrade.
- When the device is booted with an empty configuration.
- When user-defined keys/certificates are deleted.

Some commands in this section are used to generate user-defined RSA/DSA keys and certificates that replace the default keys and are used by SSL and SSH server commands. Other commands can be used to import these keys from an external source.

These keys and certificates are stored in the configuration files.

The following table describes when these keys/certificates are displayed.

File Type Displayed	Displayed in Show Command Without Detailed	Displayed in Show Command With Detailed
Startup Config	Only user-defined keys/certificates.	Option is not supported.
Running Config	Keys are not displayed.	All keys (default and user-defined)
Text-based CLI (local backup config. file, or remote backup config. file)	Keys are displayed as they were copied. There is no distinction here between default and user-defined keys.	Option is not supported.

The following table describes how keys/certificates can be copied from one type of configuration file to another (using the copy command).

Destination File Type	Copy from Running Config.	Copy from Startup Config.	Copy from Remote/Local Backup Config. File
Startup Config.	All keys/certificates are copied (but only user-defined ones can be displayed)	Option is not supported.	All keys/certificates present in this file are copied (*, **).
Running Config	N/A	Only user defined.	All keys/certificates present in this file are copied (*).
Text-based CLI (local backup config. file, or remote backup config. file)	All keys (default and user)	Only user defined.	All keys/certificates present in this file are copied (**)

* If the Running Configuration file on the device contains default keys (not user-defined ones), the same default keys remain after reboot.

** In a text-based configuration file, there is no distinction between automatically-defined, default keys and user-defined keys.

Lists of Commands

crypto key generate dsa

The crypto key generate dsa Global Configuration mode command generates a public and private DSA key (DSA key pair).

Syntax

crypto key generate dsa

Parameters

N/A

Default Configuration

The application creates a default key automatically.

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 default or user defined, a warning is displayed with a prompt to replace the existing keys with new keys.

Erasing the startup configuration or returning to factory defaults automatically deletes the default keys and they are recreated during device initialization.

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

See **Keys and Certificates** for information on how to display and copy this key pair.

Example

The following example generates a DSA key pair.

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

The SSH service is generating a private DSA key.

This may take a few minutes, depending on the key size.

.....

crypto key generate rsa

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

Syntax

`crypto key generate rsa`

Parameters

N/A

Default Configuration

The application creates a default key automatically.

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.

See **Keys and Certificates** for information on how to display and copy this key pair.

Example

The following example generates RSA key pairs where a RSA key already exists.

```
console(config)# crypto key generate rsa
Replace Existing RSA Key [y/n]? N
console(config)#
```

crypto key import

The **crypto key import** Global Configuration mode command imports the DSA/RSA key pair.

Use the **no** form of the command to remove the user key and generate a new default in its place.

Syntax

```
crypto key import {dsa | rsa}
```

```
no crypto key {dsa | rsa}
```

Parameters

N/A

Default Configuration

DSA and RSA key pairs do not exist.

Command Mode

Global Configuration mode

User Guidelines

DSA/RSA keys are imported in pairs - one public DSA/RSA key and one private DSA/RSA key.

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

This command is saved in the Running Configuration file.

Example

```
console(config)# crypto key import rsa
---- BEGIN SSH2 PRIVATE KEY ----
console(config)# encrypted crypto key import rsa
---- BEGIN SSH2 ENCRYPTED PRIVATE KEY ----
Comment: RSA Private Key
84et9C2XUfcRlpemuGINAygnLwfKkJcDM6m2OReALHScqgLhi0wMSSYNlT1IWFZP1k
EVHH
Fpt1aECZi7HfGLcp1pMZwjn1+HaXBtQjPDiEtbpScXqrg6m11/OEnwpFK2TrmUy0Ii
fwk8
E/mMfX3i/2rRZLkEBea5jrA6Q62g15naRw1ZkOges+GNeibtvZYSk1jzr56LUr6fT7
Xu5i
KMcU2b2NsuSD5yW8R/x0CW2elqDDz/biA2gSgd6FfnW2HV48bTC55eCKrsId2Mmjbe
xUdz
+RQRhzjcgMBYp6HzkD66z8HmShOU+hKd7M1K9U4Sr+Pr1vyWUJlEkOgz9O6aZoIGp4
tgm4
VDy/K/G/sI5nVL0+bR8LFUXUO/U5hohBcyRUF02fHYKZrhTiPT5Rw+Pht6/+EXKG9E
+TRs
```

```

lUADm1tCRvs+lsB33IBdvoRdd198YaA2htZay1TkbMqCUBdf10+74UOqa/b+bp67wC
YKe9
yen418MaYKtchJBQmF7sUQZQGP34VPmOMyZzon68S/ZoT77cy0ihRZx9wcIlyYhJnD
iYxP
dgXHYhW6kCTcTj6LrUSQuxCJ9su89ZIWNn5OwdgonLSpvfnabv2GHmmelaveL7JJ/7
UcfO
61q5D4PJ67Vk2xL7PqyHXN931rseTzPuJplkSLCFZ5uqTMbWWyQEKmHD10x35v1Gou
5tky
9LgIwG4d+9edctZZaggeq5cgjnsZWJgUoB4Bn4hIreyOdHDI FUPPRxkoyhGOGnJuvx
C9T9
K6BF1wBTdDQS+Gu47/0/gRoD/50q4sGkzqHsRJJ53WOT0Q1bHMTMLPpwn2nXzvfGxW
L/bu
QhZZSqRonG6MX1cP7KT7i4TPq2w2k3TGtNBnVYHx6OoNcaTHmg1N2s5OgRsyXD9tF+
+6nY
RfMN8CsV+9jQKQP7ZaGc8Ju+d72jvSwppSr032HY+Ipsz4ujkK+/X5oawZL5NnkaEQ
TQKX
RSL55S405NPOjs/pC9hg7GaVjoY2mQ7HDpSUBeTIDTlvOwC2kskA9C6aF/Axj2dXLw
eQd5
lxk7m0/mMNaiJsNk6y33LcuKjIxpNNjK9n9KzRPkGNMF0bprfenWKteDftjQ==
---- END SSH2 PRIVATE KEY ----
---- BEGIN SSH2 PUBLIC KEY ----
Comment: RSA Public Key
AAAAB3NzaC1yc2EAAAABIwAAAIEAvRHsKry6NKMkymb+yWEp9042vupLvYVq3ngt1s
B9JH
OcdK/2nw7lCQguy1mLsX8/bKMXYSk/3aBEvaoJQ82+r/nRf0y3HTy4Wp9zv0SiVC8j
LD+7
7t0aHejzfUhr0FRhWWcLnvYwr+nmrYDps6FADMC2hVA85KZRye9ifxT7ote=
---- END SSH2 PUBLIC KEY ----

```

show crypto key

The `show crypto key` Privileged EXEC mode command displays the device's SSH public keys for both default and user-defined keys.

Syntax

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

Parameters

- *mypubkey*—Displays only the public key.
- *rsa*—Displays the RSA key.
- *dsa*—Displays the DSA key.

Default Configuration

N/A

Command Mode

Privileged EXEC mode

User Guidelines

See **Keys and Certificates** for information on how to display and copy this key pair.

Example

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

```
console# show crypto key mypubkey dsa
---- BEGIN SSH2 PUBLIC KEY ----
Comment: RSA Public Key
AAAAB3NzaClyc2EAAAABIwAAAIEAzN31fu56KSE0ZdrGVPIJHpsAs8G8NDIkB
dqz2q0QPikCnLPw0Xsk9tTVKaHZQ5jJbXn81QZpolaPLJIIH3B1cc96D7IFf
VkbPbMRbz24dpuWmPVVLU1Qy5nCKdDCui5KKVD6zj3gpubLhMJor7AJAAu5e
BrIi2IuwMVJuak5M098=
---- END SSH2 PUBLIC KEY ----
Public Key Fingerprint:
6f:93:ca:01:89:6a:de:6e:ee:c5:18:82:b2:10:bc:1e
```

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*]] [**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** *length*—Regenerates SSL RSA key and specifies the SSL's RSA key length. (Range: 512–2048)

The following elements can be associated with the key. When the key is displayed, they are also displayed.

- **cn** *common-name*—Specifies the fully qualified device URL or IP address. (Length: 1–64 characters). If unspecified, defaults to the lowest IP address of the device (when the certificate is generated).
- **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 SSL's RSA key length is 1024.

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

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

If both certificates 1 and 2 have been generated, use the **ip https certificate** command to activate one of them.

See **Keys and Certificates** for information on how to display and copy this key pair.

Erasing the startup configuration or returning to factory defaults automatically deletes the default keys and they are recreated during device initialization.

Example

The following example generates a self-signed certificate for HTTPS whose length is 2048 bytes.

```
console(config)# crypto certificate 1 generate key-generate  
2048
```

crypto certificate request

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

Syntax

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

Parameters

- *number*—Specifies the certificate number. (Range: 1–2)
- The following elements can be associated with the key. When the key is displayed, they are also displayed.
 - **cn** *common-name*—Specifies the fully qualified device URL or IP address. (Length: 1–64 characters). If unspecified, defaults to the lowest IP address of the device (when the certificate is generated).
 - **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)

Default Configuration

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.

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** 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** 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-----
MIwTCCASoCAQAwYjELMAkGA1UEBhMCUFaxCzAJBgNVBAGTAkNDMQswCQYDV
QQH
EwRDEMMAoGA1UEChMDZGxkMQwwCgYDVQQLEwNkbGQxCzAJBgNVBAMTAmxkM
RAw
DgKozIhvcNAQkBFgFsMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC8e
cwQ
HdML0831i0fh/F0MV/Kib6Sz5p+3nUUenbfHp/igVPmFM+1nbqTDekb2ymC
u6K
aKvEbVLF9F2LmM7VPjDBb9bb4jnxkvwW/wzDLvW2rsy5NPmH1QVl+8Ubx3G
yCm
/oW93BSOFwxwEsp58kf+sPYPy+/8wmoNtDwIDAQABoB8wHQYJKoZIhvcNA
QkH
MRDjEyMwgICCAgICAICAQIMA0GCSqGSIb3DQEBAUAA4GBAGb8UgIx7rB05
m+2
m5ZZPhIwl8ARSPXwhVdJexFjbnmvcacqjPG8pIiRV6LkxryGF2bVU3jKEip
cZa
g+uNpyTkDt3ZVU72pjz/fa8TF0n3
-----END CERTIFICATE REQUEST-----
```

crypto certificate import

The **crypto certificate import** Global Configuration mode command imports a certificate signed by a Certification Authority for HTTPS. In addition, the RSA key-pair can also be imported.

Use the **no** form of the command to delete the user-defined keys and certificate.

Syntax

crypto certificate *number* **import**

no crypto certificate *number*

Parameters

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

Default Configuration

N/A

Command Mode

Global Configuration mode

User Guidelines

To end the session (return to the command line to enter the next command), enter a blank line.

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

If only the certificate is imported, and the public key found in the certificate does not match the device's SSL RSA key, the command fails. If both the public key and the certificate are imported, and the public key found in the certificate does not match the imported RSA key, the command fails.

This command is saved in the Running configuration file.

See **Keys and Certificates** for information on how to display and copy this key pair.

Examples

Example 1 - The following example imports a certificate signed by the Certification Authority for HTTPS.

```
console(config)# crypto certificate 1 import
Please paste the input now, add a period (.) on a separate line
after the input, and press Enter.
-----BEGIN CERTIFICATE-----
MIIBkzCB/QIBADBUMQswCQYDVQQGEwIgdIDEKMAgGA1UECBMBIDEKMAgGA1U
EBxMB
IDEVMBMGA1UEAxMMMTAuNS4yMzQuMjA5MQowCAQYDVQQKEwEgMQowCAQYDVQQ
LEwEg
```

```
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDK+beogIcke73sBSL7tC2
DMZrY
OOg9XM1AxfOiqLlQJHd4xP+BHGZWwfkjKjUDBpZn52LxdDulKrpB/h0+TZP
0Fv38
7mIDqtnoF1NLSWxkVKRM5LPka0L/halpYxp7EWA5iDBzSw5s04lv0bSN7o
aGjFA
6t4SW2rrnDy8JbwjWQIDAQABoAAwDQYJKoZIhvcNAQEEBQADgYEAuqYQiNj
st6hI
XFDxe7I8Od3Uyt3Dmf7KE/AmUV0Pif2yUluy/RuxRwKhDp/lGrK12tzLQz+
s5Ox7
Klft/IcjbzBYXLvih45ASWG3TRv2WVKyWs89rPPXu5hKxggEeTvWqpuS+gX
rIqjW
WVZd0nlfXhMacoflgnnEmweIzmrqXBs=
```

.

-----END CERTIFICATE-----

Certificate imported successfully.

Issued by : C= , ST= , L= , CN=0.0.0.0, O= , OU=

Valid From: Jan 24 18:41:24 2011 GMT

Valid to: Jan 24 18:41:24 2012 GMT

Subject: C=US , ST= , L= , CN=router.gm.com, O= General Motors, OU=

SHA1 Finger print: DC789788 DC88A988 127897BC BB789788

Example 2: The following example imports a certificate signed by the Certification Authority for HTTPS, and the RSA key-pair.

```
console(config)# crypto certificate 1 import
```

Please paste the input now, add a period (.) on a separate line after the input, and press Enter.

-----BEGIN RSA PRIVATE KEY-----

```
ACnrqImEGlXkwxBuZULAO9nHq9IGJsnkf7/MauGPVqxt5vfDf77uQ5CPf49
JWQhu07cVXh
```

```
2OwrBhJgB69vLULJu jM9p1IXFpMk8qR3NS7JzlInYAWjHKKbEZBMKSa6+t
/UzVxevKK6H
```

```
TGB7vMxi+hv1bL9zygvmQ6+/6QfqA51c4nP/8a6NjO/ZOAgvNAMKNr2Wa+t
GUOoAgL0b/C
```

```
11EoqzpCq5mT7+VOFhPSO4dUU+NwLv1YCb1Fb7MFoAa0N+y+2NwoGp0pxOv
DA9ENY17qsZ
```

```
MWmCfXu52/IxC7fD8FWxEbTks4V81Xqa7K6ET657xs7m8yTJFLZJyVawGXX
nIU56uTzhzhW
dKWwC0e/vwMgPtLlWyxWynnaP0fAJ+PawOAdsK75bo79NBim3HcNVXhWNzq
fg2s3AYCRBx
WuGoazpxHZ0s4+7swmNZtS0xI4ek43d7RaoedGK1jhPqLHuzXHUon7Zx15C
UtP3sbH1+XI
B3u4EEcEngYMewy5obn1vnFSot+d5JHuRwzEaRAIKfbHa34a1VJaN+2AMCb
0hpI3IkreYo
A8Lk6UMOuIQaMnhYf+RyPXhPOQs01PpIPHKBGti6pj39XMviyRXvSpn5+eI
YPhve5jYaEn
UeOnVZRhNcVnruJAYXSLhjApf5iIQR1JiJb/mVt8+zpqcCU9HCWQqsMrNFO
FrSpCbHu5V4
ZX4jmd9tTJ2mhekoQf1dwUZbfYkRYSK70ps8u7BtgpRfSRUr7g0LfzhzMus
woDSnB65pkC
q17yZnBeRS0zrUDgHLLRfzwjwmXjmwObxYfRGMLp4=
-----END RSA PRIVATE KEY-----
-----BEGIN RSA PUBLIC KEY-----
MIGHA0GBAMVufgfJYLbUzmbm6UoLD3ewHYd1ZMXY4A3KLF2SXUD1TIXq84a
ME8DIitSfB2
Cqy4QB5InhgAobBKC96VRsUe2rzoNG4QDkj2L9ukQOvoFBYnmbzHc7a+704
3wfVmH+QOXf
TbnRDhIMVrZJGbz1lc9IzGky1121XmicY0/nwsXDAgej
-----END RSA PUBLIC KEY-----
-----BEGIN CERTIFICATE-----
MIIBkzCB/QIBADBUMQswCQYDVQQGEwIgdEKMAGGA1UECBMBIDEKMAGGA1U
EBxMB
IDEVMBMGA1UEAxMMMTAuNS4yMzQuMjA5MzQwCAYDVQQKEwEgMQowCAYDVQQ
LEwEg
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDK+beogIcke73sBSL7tC2
DMZrY
00g9XM1Axf0iqLlQJHd4xP+BHGZwWfkjKjUDBpZn52LxdDulKrpB/h0+TZP
0Fv38
7mIDqtnoF1NLsWxkVKRM5LPka0L/ha1pYxp7EWAt5iDBzSw5s041v0bSN7o
agjFA
6t4SW2rrnDy8JbwjWQIDAQABAAAwDQYJKoZIhvcNAQEEBQADgYEAuqYQINJ
st6hI
XFDxe7I8Od3Uyt3Dmf7KE/AmUV0Pif2yUluy/RuxRwKhDp/lGrK12tzLQz+
s50x7
Klft/IcjbBYXLvih45ASWG3TRv2WVKyWs89rPPXu5hKxggEeTvWqpuS+gX
rIqjW
```

```
WVZd0n1fXhMacoflgnnEmweIzmrqXBs=
-----END CERTIFICATE-----
.
Certificate imported successfully.
Issued by : C= , ST= , L= , CN=0.0.0.0, O= , OU=
Valid From: Jan 24 18:41:24 2011 GMT
Valid to: Jan 24 18:41:24 2012 GMT
Subject: C=US , ST= , L= , CN=router.gm.com, O= General Motors, OU=
SHA1 Finger print: DC789788 DC88A988 127897BC BB789788
```

show crypto certificate

The `show crypto certificate` Privileged EXEC mode command displays the device SSL certificates and key-pair for both default and user defined keys.

Syntax

```
show crypto certificate [mycertificate] [number]
```

Parameters

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

Default Configuration

Certificate number 1.

Command Mode

Privileged EXEC mode

Examples

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

```
console# show crypto certificate 1
Certificate 1:
Certificate Source: Default
-----BEGIN CERTIFICATE-----
```

dHmUgUm9vdCBDZXJ0aWZpZXIwXDANBgkqhkiG9w0BAQEFAANLADBIAkEAp4
HS
nnH/xQSGA2ffkRBwU2XIxb7n8VPsTmlxyJ1t1l1a1GaqchfMqqe0kmfhcoHS
Wr
yflFpD0MWOTgDAwIDAQABo4IBojCCAZ4wEwYJKwYBBAGCNxQCBAYeBABDAE
Ew
CwR0PBAQDAgFGMA8GA1UdEwEB/wQFMAMBAf8wHQYDVR0OBBYEFaf4MT9BRD
47
ZvKBAEL9Ggp+6MIIBNgYDVR0fBIIBLTCCASkwgdKggc+ggcyGgclsZGFwOi
8v
L0VByb3h5JTIwU29mdHdhcmUlmjBSb290JTIwQ2VydGlmaWVyLENOPXNlcn
Zl

-----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

Web Server Commands

ip http server

Use the **ip http server** Global Configuration mode command to enable 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

Parameters

N/A

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
```

ip http port

The **ip http port** Global Configuration mode command specifies the TCP port used by the web browser interface. Use the **no** form of this command to restore the default configuration.

Syntax

```
ip http port port-number  
no ip http port
```

Parameters

port *port-number*—For use by the HTTP server. (Range: 1–65534)

Default Configuration

The default port number is 80.

Command Mode

Global Configuration mode

Example

The following example configures the http port number as 100.

```
console(config)# ip http port 100
```

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

- **idle-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

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
```

ip http secure-server

Use the **ip http secure-server** Global Configuration mode command to enable the device to be configured or monitored 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

After this command is used, you must generate a certificate using [crypto certificate generate](#). If no certificate is generated, this command has no effect.

Example

```
console(config)# ip http secure-server
```

ip http secure-port

Use the `ip http secure-port` Global Configuration mode command to specify the TCP port to be used by the secure web browser. To use the default port, use the `no` form of this command.

Syntax

```
ip http secure-port port-number
```

```
no ip http secure-port
```

Parameters

port-number—Port number for use by the HTTPS server (Range: 1–65534)

Default Configuration

The default port number is 443.

Command Mode

Global Configuration mode

Example

```
console(config)# ip http secure-port 1234
```

ip https certificate

Use the `ip https certificate` Global Configuration mode command to configure 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

First, use [crypto certificate generate](#) to generate one or two HTTPS certificates. Then use this command to specify which is the active certificate.

Example

The following example configures the active certificate for HTTPS.

```
console(config)# ip https certificate 2
```

show ip http

The `show ip http` Privileged EXEC mode command displays the HTTP server configuration.

Syntax

`show ip http`

Command Mode

Privileged EXEC mode

Example

The following example displays the HTTP server configuration.

```
console# show ip http
HTTP server enabled
Port: 80
Interactive timeout: 10 minutes
```

show ip https

The `show ip https` Privileged 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

ssl version

Use the `ssl version` Global Configuration mode command to define the version of the supported SSL.

Use the `no` form to return to the default.

Syntax

`ssl version {v2&v3 | v3}`

`no ssl version`

Parameters

- `v2&v3`—SSLv2 and SSLv3 are supported.
- `v3`—Only versions starting with SSLv3 are supported.

Defaults

`v3`

Command Modes

Global Configuration mode

Examples

```
console(config)# ssl version v3&v3
```

show ssl version

Use the `show ssl versions` Privileged EXEC mode command to display the SSL supported version.

Syntax

`show ssl version`

Parameters

N/A

Defaults

N/A

Command Modes

Privileged EXEC mode

Examples

```
console# show ssl version
```

Current supported version: SSLv2 and SSLv3

Telnet, Secure Shell and Secure Login Commands

ip telnet server

Use the `ip telnet server` Global Configuration mode command to enable the device as a Telnet server that accepts connection requests from remote Telnet clients. Remote Telnet clients can configure the device through the Telnet connections.

Use the `no` form of this command to disable the Telnet server functionality on the device.

Syntax

`ip telnet server`

`no ip telnet server`

Default Configuration

Disabled

Command Mode

Global Configuration mode

User Guidelines

The device can be enabled to accept connection requests from both remote SSH and Telnet clients. It is recommended that the remote client connects to the device using SSH (as opposed to Telnet), since SSH is a secure protocol and Telnet is not. To enable the device to be an SSH server, use the `ip ssh server` command.

Example

The following example enables the device to be configured from a Telnet server.

```
console(config)# ip telnet server
```

ip ssh server

The `ip ssh server` Global Configuration mode command enables the device to be an SSH server and so to accept connection requests from remote SSH clients. Remote SSH clients can manage the device through the SSH connection.

Use the `no` form of this command to disable the SSH server functionality from the device.

Syntax

```
ip ssh server
```

```
no ip ssh server
```

Default Configuration

The SSH server functionality is disabled by default.

Command Mode

Global Configuration mode

User Guidelines

The device, as an SSH server, generates the encryption keys automatically.

To generate new SSH server keys, use the `crypto key generate dsa` and `crypto key generate rsa` commands.

Example

The following example enables configuring the device to be an SSH server.

```
console(config)# ip ssh server
```

ip ssh port

The `ip ssh port` Global Configuration mode command specifies the TCP port used by the SSH server. Use the `no` form of this command to restore the default configuration.

Syntax

`ip ssh port port-number`

`no ip ssh port`

Parameters

- *port-number*—Specifies the TCP port number to be used by the SSH server. (Range: 1–65535).

Default Configuration

The default TCP port number is 22.

Command Mode

Global Configuration mode

Example

The following example specifies that TCP port number 8080 is used by the SSH server.

```
console(config)# ip ssh port 8080
```

ip ssh password-auth

Use the `ip ssh password-auth` Global Configuration mode command to enable password authentication of incoming SSH sessions.

Use the `no` form of this command to disable this function.

Syntax

`ip ssh password-auth`

`no ip ssh password-auth`

Default Configuration

Password authentication of incoming SSH sessions is disabled.

Command Mode

Global Configuration mode

User Guidelines

This command enables password key authentication by a local SSH server of remote SSH clients.

The local SSH server advertises all enabled SSH authentication methods and remote SSH clients are responsible for choosing one of them.

After a remote SSH client is successfully authenticated by public key, the client must still be AAA-authenticated to gain management access to the device.

If no SSH authentication method is enabled, remote SSH clients must still be AAA-authenticated before being granted management access to the device.

Example

The following example enables password authentication of the SSH client.

```
console(config)# ip ssh password-auth
```

ip ssh pubkey-auth

Use the `ip ssh pubkey-auth` Global Configuration mode command to enable public key authentication of incoming SSH sessions.

Use the `no` form of this command to disable this function.

Syntax

```
ip ssh pubkey-auth [auto-login]
```

```
no ip ssh pubkey-auth
```

Parameters

- **auto-login**—Specifies that the device management AAA authentication (CLI login) is not needed. By default, the login is required after the SSH authentication.

Default Configuration

Public key authentication of incoming SSH sessions is disabled.

Command Mode

Global Configuration mode

User Guidelines

This command enables public key authentication by a local SSH server of remote SSH clients.

The local SSH server advertises all enabled SSH authentication methods and remote SSH clients are responsible for choosing one of them.

After a remote SSH client is successfully authenticated by public key, the client must still be AAA-authenticated to gain management access to the device, except if the **auto-login** parameter was specified.

If no SSH authentication method is enabled, remote SSH clients must still be AAA-authenticated before being granted management access to the device.

If the **auto-login** keyword is specified for SSH authentication by public key management access is granted if SSH authentication succeeds and the name of SSH used is found in the local user database. The device management AAA authentication is transparent to the user. If the user name is not in the local user database, then the user receives a warning message, and the user will need to pass the device management AAA authentication independently of the SSH authentication.

if the **auto-login** keyword is not specified, management access is granted only if the user engages and passes both SSH authentication and device management AAA authentication independently. If no SSH authentication method is enabled management access is granted only if the user is AAA authenticated by the device management. No SSH authentication method means SSH is enabled and neither SSH authentication by public key nor password is enabled.

Example

The following example enables authentication of the SSH client.

```
console(config)# ip ssh pubkey-auth
```

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 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 the user 'bob'.

```
console(config)# crypto key pubkey-chain ssh
console(config-keychain)# user-key bob rsa
console(config-keychain-key)# key-string
AAAAB3NzaC1yc2EAAAADAQABAAQACvTnRwPWl
Al4kpgIw9GBRonZQZxjHKcqKL6rMlQ+
ZNXfZSkvHG+QusIZ/76ILmFT34v7u7ChFAE+
```

```
Vu4GRfpSwoQUvV35LqJJK67IOU/zfwO1lg
kTwm175QR9gHujS6KwGN2QWXgh3ub8gDjTSq
muSn/Wd05iDX2IExQWu08licglk02LYciz
+Z4TrEU/9FJxwPiVQOjc+KBXuR0juNg5nFYsY
0ZCk0N/W9a/tnkmlshRE7Di71+w3fNiOA
6w9o44t6+AINEICBCCA4YcF6zMzaTlweFwWx6f+
Rmt5nhhqdAtN/4oJfce166DqVX1gWmN
zNR4DYDvSzg0lDnwCAC8Qh
Fingerprint:
a4:16:46:23:5a:8d:1d:b5:37:59:eb:44:13:b9:33:e9
```

user-key

The **user-key** SSH Public Key-string Configuration mode command associates a username with a manually-configured SSH public key.

Use the **no user-key** command to remove an SSH user and the associated 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

After entering this command, the existing key, if any, associated with the user will be deleted. You must follow this command with the `key-string` command to configure the key to the user.

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-keychain)# user-key bob rsa
console(config-keychain-key)# key-string row
AAAAB3NzaC1yc2EAAAADAQABAAQACvTnRwPW1
```

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. The maximum length of a row is 160 characters.
- `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.

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-keychain)# user-key bob rsa
console(config-keychain-key)# key-string
AAAAB3NzaClyc2EAAAADAQABAAQACvTnRwPWl
Al4kpgIw9GBRonZQZxjHKcqKL6rMlQ+
ZNXfZSkvHG+QusIZ/76ILmFT34v7u7ChFAE+
Vu4GRfpSwoQUvV35LqJjk67IOU/zfwO1lg
kTwm175QR9gHujS6KwGN2QWXgh3ub8gDjTSq
muSn/Wd05iDX2IExQWu08licglk02LYciz
+Z4TrEU/9FJxwPiVQOjc+KBXuR0juNg5nFYsY
0Zck0N/W9a/tnkmlshRE7Di71+w3fNiOA
6w9o44t6+AINEICBCCA4YcF6zMzaT1wefWwX6f+
Rmt5nhhqdatN/4oJfce166DqVXlgWmN
zNR4DYDvsZsg0ldnwCAC8Qh
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-keychain)# user-key bob rsa
console(config-keychain-key)# key-string row AAAAB3Nza
console(config-keychain-key)# key-string row Clyc2
```

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 with auto-login.
SSH Password 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
182.20.2.1	Bob Smith	1.5	3DES	Password

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

Field	Description
IP Address	The client address
SSH Username	The user name
Version	The SSH version number
Cipher	The encryption type (3DES, Blowfish, RC4)

Field	Description
Auth Code	The authentication Code (HMAC-MD5, HMAC-SHA1) or Password

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      Fingerprint
```

```
-----  
bob          9A:CC:01:C5:78:39:27:86:79:CC:23:C5:98:59:F1:86  
john        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      Fingerprint  
-----  
bob          9A:CC:01:C5:78:39:27:86:79:CC:23:C5:98:59:F1:86
```

Line Commands

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 terminal line mode.
- **telnet**—Configures the device as a virtual terminal for remote access (Telnet).
- **ssh**—Configures the device as a virtual terminal for secured remote access (SSH).

Command Mode

Global Configuration mode

Example

The following example configures the device as a virtual terminal for remote (Telnet) access.

```
console(config)# line telnet
console(config-line)#
```

speed

Use the **speed** command in Line Configuration mode to set 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 4800, 9600, 19200, 38400, 57600, and 115200.

Default Configuration

The default speed is 115200 bps.

Command Mode

Line Configuration Mode

User Guidelines

The configured speed is only 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
```

autobaud

Use the **autobaud** command in Line Configuration mode to configure 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

When this command is enabled, it is activated as follows: connect the console to the device and press the **Enter** key twice. The device detects the baud rate automatically.

Example

The following example enables autobaud.

```
console(config)# line console
console(config-line)# autobaud
```

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 telnet session idle time interval before automatic logoff to 20 minutes and 10 seconds.

```
console(config)# line telnet  
console(config-line)# exec-timeout 20 10
```

show line

The `show line` Privileged 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

Privileged 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

Authentication, Authorization and Accounting

aaa authentication login

Use the `aaa authentication login` Global Configuration mode command to set one or more authentication methods to be applied during login. Use the `no` form of this command to restore the default authentication method.

Syntax

```
aaa authentication login {default / 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 (this list is unnamed).
- *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). Each additional authentication method is 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
<code>enable</code>	Uses the enable password for authentication.
<code>line</code>	Uses the line password for authentication.
<code>local</code>	Uses the locally-defined usernames for authentication.
<code>none</code>	Uses no authentication.
<code>radius</code>	Uses the list of all RADIUS servers for authentication.

Keyword	Description
tacacs	Uses the list of all TACACS+ servers for authentication.

Default Configuration

If no methods are specified, the default are the locally-defined users and passwords. This is the same as entering the command **aaa authentication login local**.

Command Mode

Global Configuration mode

User Guidelines

Create a list of authentication methods by entering this command with the *list-name* parameter where *list-name* is any character string. 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 the **login authentication** command.

no aaa authentication login list-name deletes a list-name only if it has not been referenced by another command.

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
```

aaa authentication enable

The **aaa authentication enable** Global Configuration mode command sets one or more authentication methods for accessing higher privilege levels. 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
enable	Uses the enable password for authentication.
line	Uses the line password for authentication.
none	Uses no authentication.
radius	Uses the list of all RADIUS servers for authentication.
tacacs	Uses the list of all TACACS+ servers for authentication.

Default Configuration

The **enable password** command defines 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 by this command are used with the **enable authentication** command.

All **aaa authentication enable** requests sent by the device to a RADIUS server include the username \$enabx\$, where x is the requested privilege level.

All **aaa authentication enable** requests sent by the device to a TACACS+ server include the username that is entered for login authentication.

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
```

login authentication

The **login authentication** Line Configuration mode command specifies the login authentication method list for a remote Telnet or console session. Use the **no** form of this command to restore the default authentication method.

Syntax

```
login authentication {default / list-name}
```

no login authentication

Parameters

- **default**—Uses the default list created with the **aaa authentication login** command.
- *list-name*—Uses the specified list created with the **aaa authentication login** command.

Default Configuration

default

Command Mode

Line Configuration Mode

Examples

Example 1 - The following example specifies the login authentication method as the default method for a console session.

```
console(config)# line console
console(config-line)# login authentication default
```

Example 2 - The following example sets the authentication login methods for the console as a list of methods.

```
console(config)# aaa authentication login authen-list radius
local none
console(config)# line console
console(config-line)# login authentication authen-list
```

enable authentication

The **enable authentication** Line Configuration mode command specifies the authentication method for accessing a higher privilege level from a remote Telnet or console. Use the **no** form of this command to restore the default authentication method.

Syntax

`enable authentication {default / list-name}`

`no enable authentication`

Parameters

- **default**—Uses the default list created with the `aaa authentication enable` command.
- *list-name*—Uses the specified list created with the `aaa authentication enable` command.

Default Configuration

default.

Command Mode

Line Configuration Mode

Example

Example 1 - The following example specifies the authentication method as the default method when accessing a higher privilege level from a console.

```
console(config)# line console
console(config-line)# enable authentication default
```

Example 2 - The following example sets a list of authentication methods 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
```

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
local	Uses the local username database for authentication.
none	Uses no authentication.
radius	Uses the list of all RADIUS servers for authentication.
tacacs	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.

Example

The following example specifies the HTTP access authentication methods.

```
console(config)# ip http authentication aaa login-  
authentication radius local none
```

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_Login           Console_Enable
Telnet              Default                 Default
SSH                 Default                 Default

HTTP, HTTPS: Radius, local
Dot1x: Radius
```

password

Use the **password** Line Configuration mode command to specify a password on a line (also known as an 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
```

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.

Syntax

```
enable password {unencrypted-password / encrypted encrypted-password}
no enable password
```

Parameters

- *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

Passwords are encrypted by default.

Command Mode

Global Configuration mode

User Guidelines

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.

Passwords are encrypted by default. You only are required to use the **encrypted** keyword when you are actually entering an encrypted keyword.

Examples

Example 1 - The command sets a password 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)# enable password encrypted  
4b529f21c93d4706090285b0c10172eb073ffe4
```

service password-recovery

Use the **service password-recovery** Global Configuration mode command to enable the password-recovery mechanism. This mechanism allows an end user, with physical access to the console port of the device, to enter the boot menu and trigger the password recovery process. Use the **no service password-recovery** command to disable the password-recovery mechanism. When the password-recovery mechanism is disabled, accessing the boot menu is still allowed and the user can trigger the password recovery process. The difference is, that in this case, all the configuration files and all the user files are removed. The following log message is generated to the terminal: "All the configuration and user files were removed".

Syntax

service password-recovery

no service password-recovery

Parameters

N/A

Default Configuration

The service password recovery is enabled by default.

Command Mode

Global Configuration mode

User Guidelines

- If password recovery is enabled, the user can access the boot menu and trigger the password recovery in the boot menu. All configuration files and user files are kept.
- If password recovery is disabled, the user can access the boot menu and trigger the password recovery in the boot menu. The configuration files and user files are removed.
- If a device is configured to protect its sensitive data with a user-defined passphrase for (Secure Sensitive Data), then the user cannot trigger the password recovery from the boot menu even if password recovery is enabled.
- If a device is configured to protect its sensitive data with a user-defined passphrase for (Secure Sensitive Data), then the user cannot trigger the password recovery from the boot menu even if password recovery is enabled.

Example

The following command disables password recovery:

```
console(config)# no service password recovery
```

Note that choosing to use Password recovery option in the Boot Menu during the boot process will remove the configuration files and the user files. Would you like to continue ? Y/N.

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 {unencrypted-password /  
{encrypted encrypted-password}}
```

```
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.
- **password**—Specifies the password for this username. (Range: 1–64)
- *unencrypted-password*—The authentication password for the user. (Range: 1–159)
- **encrypted *encrypted-password***—Specifies that the password is MD5 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

No user is defined.

Command Mode

Global Configuration mode

Usage Guidelines

The last user (regardless of whether it is the default user or any user) cannot be removed and cannot be a remote user.

Examples

Example 1 - Sets an unencrypted password for user tom (level 15). It will be encrypted in the configuration file.

```
console(config)# username tom password 1234
```

show users accounts

The `show users accounts` Privileged EXEC mode command displays information about the users local database.

Syntax

`show users 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 users accounts
```

Username	Password	Expiry date
-----	-----	-----
Bob		Jan 18 2005
Robert		Jan 19 2005
Smith		

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

Field	Description
Username	The user name.
Password Expiry date	The user's password expiration date.

aaa accounting login

Use the **aaa accounting login** command in Global Configuration mode to enable accounting of device management sessions. Use the **no** form of this command to disable accounting.

Syntax

```
aaa accounting login start-stop group {radius | tacacs+}
```

```
no aaa accounting login start-stop [group {radius | tacacs+}]
```

Parameters

- **group radius**—Uses a RADIUS server for accounting.
- **group tacacs+**—Uses a TACACS+ server for accounting.

Default Configuration

Disabled

Command Mode

Global Configuration mode

User Guidelines

This command enables the recording of device management sessions (Telnet, serial and WEB but not SNMP).

It records only users that were identified with a username (e.g. a user that was logged in with a line password is not recorded).

If accounting is activated, the device sends a “start”/“stop” messages to a RADIUS server when a user logs in / logs out respectively.

The device uses the configured priorities of the available RADIUS/TACACS+ servers in order to select the RADIUS/TACACS+ server.

The following table describes the supported RADIUS accounting attributes values, and in which messages they are sent by the switch.

Name	Start Message	Stop Message	Description
User-Name (1)	Yes	Yes	User's identity.
NAS-IP-Address (4)	Yes	Yes	The switch IP address that is used for the session with the RADIUS server.
Class (25)	Yes	Yes	Arbitrary value is included in all accounting packets for a specific session.
Called-Station-ID (30)	Yes	Yes	The switch IP address that is used for the management session.
Calling-Station-ID (31)	Yes	Yes	The user IP address.
Acct-Session-ID (44)	Yes	Yes	A unique accounting identifier.
Acct-Authentic (45)	Yes	Yes	Indicates how the supplicant was authenticated.
Acct-Session-Time (46)	No	Yes	Indicates how long the user was logged in.
Acct-Terminate-Cause (49)	No	Yes	Reports why the session was terminated.

The following table describes the supported TACACS+ accounting arguments and in which messages they are sent by the switch.

Name	Description	Start Message	Stop Message
task_id	A unique accounting session identifier.	Yes	Yes
user	username that is entered for login authentication	Yes	Yes

Name	Description	Start Message	Stop Message
rem-addr	IP address.of the user	Yes	Yes
elapsed-time	Indicates how long the user was logged in.	No	Yes
reason	Reports why the session was terminated.	No	Yes

Example

```
console(config)# aaa accounting login start-stop group radius
```

aaa accounting dot1x

To enable accounting of 802.1x sessions, use the **aaa accounting dot1x** Global Configuration mode command. Use the **no** form of this command to disable accounting.

Syntax

```
aaa accounting dot1x start-stop group radius
```

```
no aaa accounting dot1x start-stop group radius
```

Parameters

N/A

Default Configuration

Disabled

Command Mode

Global Configuration mode

User Guidelines

This command enables the recording of 802.1x sessions.

If accounting is activated, the device sends start/stop messages to a RADIUS server when a user logs in / logs out to the network, respectively.

The device uses the configured priorities of the available RADIUS servers in order to select the RADIUS server.

If a new supplicant replaces an old supplicant (even if the port state remains authorized), the software sends a stop message for the old supplicant and a start message for the new supplicant.

In multiple sessions mode (dot1x multiple-hosts authentication), the software sends start/stop messages for each authenticated supplicant.

In multiple hosts mode (dot1x multiple-hosts), the software sends start/stop messages only for the supplicant that has been authenticated.

The software does not send start/stop messages if the port is force-authorized.

The software does not send start/stop messages for hosts that are sending traffic on the guest VLAN or on the unauthenticated VLANs.

The following table describes the supported Radius accounting Attributes Values and when they are sent by the switch.

Name	Start	Stop	Description
User-Name (1)	Yes	Yes	Supplicant's identity.
NAS-IP-Address (4)	Yes	Yes	The switch IP address that is used for the session with the RADIUS server.
NAS-Port (5)	Yes	Yes	The switch port from where the supplicant has logged in.
Class (25)	Yes	Yes	The arbitrary value that is included in all accounting packets for a specific session.
Called-Station-ID (30)	Yes	Yes	The switch MAC address.
Calling-Station-ID (31)	Yes	Yes	The supplicant MAC address.
Acct-Session-ID (44)	Yes	Yes	A unique accounting identifier.
Acct-Authentic (45)	Yes	Yes	Indicates how the supplicant was authenticated.
Acct-Session-Time (46)	No	Yes	Indicates how long the supplicant was logged in.
Acct-Terminate-Cause (49)	No	Yes	Reports why the session was terminated.

Name	Start	Stop	Description
Nas-Port-Type (61)	Yes	Yes	Indicates the supplicant physical port type.

Example

```
console(config)# aaa accounting dot1x start-stop group radius
```

show accounting

The `show accounting` EXEC mode command displays information as to which type of accounting is enabled on the switch.

Syntax

```
show accounting
```

Parameters

N/A

Default Configuration

N/A

Command Mode

User EXEC mode

Example

The following example displays information about the accounting status.

```
console# show accounting
Login: Radius
802.1x: Disabled
```

passwords min-length

To configure the minimal length required for passwords in the local database, use the `passwords min-length` command in Global Configuration mode. Use the `no` form to remove the requirement.

Syntax

`passwords min-length length`

`no passwords min-length`

Parameters

- *length*—Specifies the minimal length required for passwords. (Range: 8-64)

Default Configuration

There is no minimal length requirement until this command is executed.

Command Mode

Global Configuration mode

User Guidelines

The setting is relevant to local user passwords, line passwords, and enable passwords.

The software checks the minimum length requirement when defining a password in an unencrypted format, or when a user tries to log in.

Note that if a password is inserted in encrypted format, the minimum length requirement is checked during user login only.

Passwords that were defined before defining the minimum length requirement are only checked during user login.

Example

The following example configures the minimal required password length to 8 characters.

```
console(config)# passwords min-length 8
```

passwords strength-check enable

Use the `passwords strength-check enable` Global Configuration mode command to enforce minimum password strength. The `no` form of this command disables enforcing password strength.

Syntax

`passwords strength-check enable`

`no passwords strength-check enable`

Parameters

N/A

Default Configuration

Disabled

Command Mode

Global Configuration mode

User Guidelines

If password strength is enabled, the user is forced to enter passwords that:

- Contain characters from user-defined several character classes (uppercase letters, lowercase letters, numbers, and special characters available on a standard keyboard).
- Contain no character that is repeated more than user-defined times consecutively.

The user can control the above attributes of password strength with specific commands.

Example

The following example enables password strength and configures the character classes to 3.

```
console(config)# passwords strength-check enable
```

passwords strength minimum character-classes

Use the `passwords strength minimum character-classes` Global Configuration mode command to configure the minimal classes required for passwords in the local database. Use the `no` form to remove the requirement.

Syntax

`passwords strength minimum character-classes number`

`no passwords strength minimum character-classes`

Parameters

- *number*—The minimal number of different character classes required in a password (Range: 0–4)

Default Configuration

0

Command Mode

Global Configuration mode

User Guidelines

The setting is relevant to local users' passwords, line passwords and enable passwords.

The software checks the minimum length requirement when you define a password in an unencrypted format.

The classes are: upper case letters, lower case letters, numbers and special characters.

passwords strength max-limit repeated-characters

Use the `passwords strength max-limit repeated-characters` Global Configuration mode command to configure the maximum number of characters in the new password that can be repeated consecutively. Use the `no` form to remove the requirement.

Syntax

passwords strength max-limit repeated-characters *number*

no passwords strength max-limit repeated-characters

Parameters

- *number*—The maximum number of characters in the new password that can be repeated consecutively. (Range: 0–16).

Default Configuration

1

Command Mode

Global Configuration mode

User Guidelines

The setting is relevant to local users' passwords, line passwords and enable passwords. The software checks the maximum number of characters in the new password that can be repeated consecutively.

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

180

Command Mode

Global Configuration mode

User Guidelines

Aging is relevant only to users of the local database and to enable a password.

To disable password aging, use `passwords aging 0`.

Using `no passwords aging` sets the aging time to the default.

Example

The following example configures the aging time to be 24 days.

```
console(config)# passwords aging 24
```

passwords history

The `passwords history` Global Configuration mode command configures the number of password changes required before a password can be reused. Use the `no` form of this command to remove the requirement.

Syntax

`passwords history number`

`no passwords history`

Parameters

- *number*—Specifies the number of password changes required before a password can be reused. (Range: 1–10).

Default Configuration

Password history is disabled.

Command Mode

Global Configuration mode

User Guidelines

The setting is relevant to local users' passwords, line passwords and enable passwords.

Password history is not checked during a configuration download.

The password history is kept even if the password history check is disabled.

The password history for a user is kept as long as the user is defined.

Example

The following example sets the number of password changes required before a password can be reused to 10.

```
console(config)# passwords history 10
```

passwords history hold-time

The `passwords history hold-time` Global Configuration mode command configures the duration that a password is relevant for tracking passwords history. Use the `no` form of this command to return to the default configuration.

Syntax

`passwords history hold-time days`

`no passwords history hold-time`

Parameters

- *days*—Specifies the number of days a password is relevant for tracking passwords history. (Range: 1–365).

Default Configuration

There is no history hold time by default.

Command Mode

Global Configuration mode

User Guidelines

The setting is relevant to local users' passwords, line passwords and enable passwords.

The passwords are not deleted from the history database when they are not relevant for the password history tracking. Increasing the hold time might "return back" passwords.

Example

The following example configures the duration that a password is relevant for tracking passwords history.

```
console(config)# passwords history hold-time 10
```

passwords lockout

The `passwords lockout` Global Configuration mode command enables user account lockout after a series of authentication failures. Use the `no` form of this command to disable the lockout feature.

Syntax

`passwords lockout` *number*

`no passwords lockout`

Parameters

- *number*—Specifies the number of authentication failures before the user account is locked-out. (Range: 1–5).

Default Configuration

Lockout is disabled.

Command Mode

Global Configuration mode

User Guidelines

The setting is relevant to local users' passwords, line passwords and enable passwords.

The account is not locked out for access from the local console.

A user can release accounts that are locked out by using the **set username active**, **set enable-password active** and **set line active** Privileged EXEC mode commands.

Disabling lockout unlocks all users.

Re-enabling lockout resets the authentication failures counters.

Changing the authentication failures threshold does not reset the counters.

Example

The following example enables user account lockout after 3 successive authentication failures.

```
console(config)# passwords lockout 3
```

aaa login-history file

The **aaa login-history file** Global Configuration mode command enables writing to the login history file. Use the **no** form of this command to disable writing to the login history file.

Syntax

aaa login-history file

no aaa login-history file

Default Configuration

Writing to the login history file is enabled.

Command Mode

Global Configuration mode

User Guidelines

The login history is stored in the device internal buffer.

Example

The following example enables writing to the login history file.

```
console(config)# aaa login-history file
```

set username active

The `set username active` Privileged EXEC mode command reactivates a locked out user account.

Syntax

```
set username name active
```

Parameters

- *name*—Specifies the user name: (Length: 1–20 characters).

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example reactivates user ‘Bob’.

```
console# set username Bob active
```

set line active

The `set line active` Privileged EXEC mode command reactivates a locked out line.

Syntax

```
set line {console | telnet | ssh} active
```

Parameters

- **console**—Reactivates the console terminal line.
- **telnet**—Reactivates the virtual terminal for remote (Telnet) console access.
- **ssh**—Reactivates the virtual terminal for secured remote (SSH) console access.

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example reactivates the virtual terminal for remote (Telnet) console access.

```
console# set line telnet active
```

set enable-password active

The `set enable-password active` Privileged EXEC mode command reactivates a locked out local password.

Syntax

```
set enable-password 15 active
```

Parameters

N/A

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example reactivates a local password.

```
console# set enable-password 15 active
```

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 strength-check is enabled
Minimal length: 8
Minimum character classes: 4
Maximal number of repeated characters: 2
Enable Passwords
Line Passwords
Line
-----
Console
Telnet
```

show users login-history

The `show users login-history` Privileged EXEC mode command displays information about the user's login history.

Syntax

`show users login-history [username name]`

Parameters

- *name*—Name of the user. (Range: 1–20 characters).

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example displays information about the users' login history.

```
console# show users login-history
File save: Enabled.

Login Time                Username      Protocol      Location
-----
Jan 18 2004 23:58:17      Robert       HTTP          172.16.1.8
Jan 19 2004 07:59:23      Robert       HTTP          172.16.0.8
Jan 19 2004 08:23:48      Bob          Serial
Jan 19 2004 08:29:29      Robert       HTTP          172.16.0.8
Jan 19 2004 08:42:31      John         SSH           172.16.0.1
Jan 19 2004 08:49:52      Betty        Telnet        172.16.1.7
```


RADIUS Commands

radius-server host

Use the `radius-server host` Global Configuration mode command to configure a RADIUS server host. Use the `no` form of the command to delete the specified RADIUS server host.

Syntax

```
radius-server host {ip-address / hostname} [auth-port auth-port-number]  
[acct-port acct-port-number] [timeout timeout] [retransmit retries]  
[deadtime deadtime] [key key-string] [source source-ip] [priority priority]  
[usage {login / dot1.x / all}]
```

```
no radius-server host {ip-address | hostname}
```

Parameters

- *ip-address*—Specifies the RADIUS server host IP address. The IP address can be an IPv4, IPv6 or IPv6z address. See [IPv6 Address Conventions](#).
- *hostname*—Specifies the RADIUS server host name. Translation to IPv4 addresses only is supported. (Length: 1–158 characters. Maximum label length of each part of the hostname: 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)
- **acct-port** *acct-port-number*—Port number for accounting requests. The host is not used for accountings if set to 0. If unspecified, the port number defaults to 1813.
- **timeout** *timeout*—Specifies the timeout value in seconds. (Range: 1–30)
- **retransmit** *retries*—Specifies the number of retry retransmissions (Range: 1–15)
- **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). If this parameter is omitted, the globally-configured radius key will be used.
- **priority** *priority*—Specifies the order in which servers are used, where 0 has the highest priority. (Range: 0–65535)
- **usage** {**login** | **dot1.x** | **all**}—Specifies the RADIUS server usage type. The possible values are:
 - **login**—Specifies that the RADIUS server is used for user login parameters authentication.
 - **dot1.x**—Specifies that the RADIUS server is used for 802.1x port authentication.
 - **all**—Specifies that the RADIUS server is used for user login authentication and 802.1x port authentication.

Default Configuration

The default authentication port number is 1812.

If **timeout** is not specified, the global value (set in the [radius-server timeout](#) command) is used.

If **retransmit** is not specified, the global value (set in the [radius-server retransmit](#) command) is used.

If **key-string** is not specified, the global value (set in the [radius-server key](#) command) is used.

If the **usage** keyword is not specified, the **all** argument is applied.

Command Mode

Global Configuration mode

User Guidelines

To specify multiple hosts, this command is used for each host.

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
```

radius-server key

Use the **radius-server key** Global Configuration mode command to set the authentication key for 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 key for all RADIUS communications between the device and the RADIUS daemon.

```
console(config)# radius-server key enterprise-server
```

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 number of retry retransmissions (Range: 1–15).

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
```

radius-server host source-interface

Use the `radius-server host source-interface` Global Configuration mode command to specify the source interface whose IPv4 address will be used as the Source IPv4 address for communication with IPv4 RADIUS servers. Use the `no` form of this command to restore the default configuration.

Syntax

`radius-server host source-interface interface-id`

no radius-server host source-interface

Parameters

- *interface-id*—Specifies the source interface.

Default Configuration

The source IPv4 address is the IPv4 address defined on the outgoing interface and belonging to next hop IPv4 subnet.

Command Mode

Global Configuration mode

User Guidelines

If the source interface is the outgoing interface, the interface IP address belonging to next hop IPv4 subnet is applied.

If the source interface is not the outgoing interface, the minimal IPv4 address defined on the source interface is applied.

If there is no available IPv4 source address, a SYSLOG message is issued when attempting to communicate with an IPv4 RADIUS server.

Example

The following example configures the VLAN 10 as the source interface.

```
console(config)# radius-server host source-interface vlan 100
```

radius-server host source-interface-ipv6

Use the **radius-server host source-interface-ipv6** Global Configuration mode command to specify the source interface whose IPv6 address will be used as the source IPv6 address for communication with IPv6 RADIUS servers. Use the **no** form of this command to restore the default configuration.

Syntax

radius-server host source-interface-ipv6 *interface-id*

no radius-server host source-interface-ipv6

Parameters

- *interface-id*—Specifies the source interface.

Default Configuration

The IPv6 source address is the IPv6 address defined on the outgoing interface and selected in accordance with RFC6724.

Command Mode

Global Configuration mode

User Guidelines

If the source interface is the outgoing interface, the source IPv6 address is an IPv6 address defined on the interfaces and selected in accordance with RFC 6724.

If the source interface is not the outgoing interface, the source IPv6 address is the minimal IPv6 address defined on the source interface and matched to the scope of the destination IPv6 address is applied.

If there is no available source IPv6 address, a SYSLOG message is issued when attempting to communicate with an IPv6 RADIUS server.

Example

The following example configures the VLAN 10 as the source interface.

```
console(config)# radius-server host source-interface-ipv6  
vlan 100
```

radius-server timeout

Use the **radius-server timeout** Global Configuration mode command to set how long 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
```

radius-server deadtime

Use the **radius-server deadtime** Global Configuration mode command to configure how long 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
```

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
IP address  Port Port Time           Dead
           Auth Acc  Out  Retransmission time  Priority Usage
-----
172.16.1.1  1812 1813 125  Global          Global 1    All
172.16.1.2  1812 1813 102  8               Global 2    AllGlobal
values
-----
TimeOut: 3
Retransmit: 3
Deadtime: 0
Source IPv4 interface: vlan 120
Source IPv6 interface: vlan 10
```

show radius-servers key

Use the `show radius-servers key` Privileged EXEC mode command to display the RADIUS server key settings.

Syntax

`show radius-servers key`

Command Mode

Privileged EXEC mode

Example

The following example displays RADIUS server key settings

```
console# show radius-servers key
```

```
IP address      Key
-----
172.16.1.1     Sharon123
172.16.1.2     Bruce123
```

```
Global key
-----
Alice456
```


TACACS+ Commands

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] [priority priority]
```

```
no tacacs-server host {ip-address | hostname}
```

Parameters

- **host** *ip-address*—Specifies the TACACS+ server host IP address. The IP address can be an IPv4, IPv6 or IPv6z address.
- **host** *hostname*—Specifies the TACACS+ server host name. (Length: 1-158 characters. Maximum label length of each part of the host name: 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 TACACS server TCP 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). If this parameter is omitted, the globally-defined key (set in [tacacs-server key](#)) will be used.

- **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 1812.

If **timeout** is not specified, the global value (set in the **tacacs-server timeout** command) is used.

If **key-string** is not specified, the global value (set in the **tacacs-server key** command) is used.

If the **source** value is not specified, the global value is set in: **tacacs-server host source-interface** or: **tacacs-server host source-interface-ipv6**.

Command Mode

Global Configuration mode

User Guidelines

Multiple **tacacs-server host** commands can be used to specify multiple hosts.

Example

The following example specifies a TACACS+ host.

```
console(config)# tacacs-server host 172.16.1.1
```

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 key for all TACACS+ servers.

```
console(config)# tacacs-server key enterprise
```

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
```

tacacs-server host source-interface

Use the `tacacs-server host source-interface` Global Configuration mode command to specify the source interface which IPv4 address will be used as the Source IPv4 address for communication with IPv4 TACACS+ servers. Use the `no` form of this command to restore the default configuration.

Syntax

`tacacs-server host source-interface interface-id`

`no tacacs-server host source-interface`

Parameters

- *interface-id*—Specifies the source interface.

Default Configuration

The source IPv4 address is the IPv4 address defined on the outgoing interface and belonging to next hop IPv4 subnet.

Command Mode

Global Configuration mode

User Guidelines

If the source interface is the outgoing interface, the interface IP address belonging to next hop IPv4 subnet is applied.

If the source interface is not the outgoing interface, the minimal IPv4 address defined on the source interface is applied.

If there is no available IPv4 source address, a SYSLOG message is issued when attempting to communicate with an IPv4 TACACS+ server.

Example

The following example configures the VLAN 10 as the source interface.

```
console(config)# tacacs-server host source-interface vlan 100
```

tacacs-server host source-interface-ipv6

Use the **tacacs-server host source-interface-ipv6** Global Configuration mode command to specify the source interface whose IPv6 address will be used as the Source IPv6 address for communication with IPv6 TACACS+ servers. Use the **no** form of this command to restore the default configuration.

Syntax

```
tacacs-server host source-interface-ipv6 interface-id
```

```
no tacacs-server host source-interface-ipv6
```

Parameters

- *interface-id*—Specifies the source interface.

Default Configuration

The IPv6 source address is the IPv6 address defined on the outgoing interface and selected in accordance with RFC6724.

Command Mode

Global Configuration mode

User Guidelines

If the source interface is the outgoing interface, the source IPv6 address is an IPv6 address defined on the interfaces and selected in accordance with RFC 6724.

If the source interface is not the outgoing interface, the source IPv6 address is the minimal IPv6 address defined on the source interface and matched to the scope of the destination IPv6 address is applied.

If there is no available source IPv6 address, a SYSLOG message is issued when attempting to communicate with an IPv6 TACACS+ server.

Example

The following example configures the VLAN 10 as the source interface.

```
console(config)# tacacs-server host source-interface-ipv6
vlan 100
```

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, IPv4 or IPv6 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      TimeOut Priority
          Connection
-----
172.16.1.1 Connected 49   No           Global  1
Global values
-----
Time Out: 3
```


SYSLOG Commands

logging on

Use the **logging on** Global Configuration mode command to enable 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
```

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 {ip-address / ipv6-address / hostname} [port port] [severity level] [facility facility] [description text]
```

```
no logging host {ipv4-address / ipv6-address / hostname}
```

Parameters

- **ip-address**—IP address of the host to be used as a SYSLOG server. The IP address can be an IPv4, IPv6 or Ipv6z address. See [IPv6 Address Conventions](#).
- **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 for each part of the host name: 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.

Examples

```
console(config)# logging host 1.1.1.121
```

```
console(config)# logging host 3000::100/SYSLOG1
```

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
```

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 buffer. (Range: 20–1000)
- **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 1000.

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 and severity level **informational**.

```
console(config)# logging buffered debugging
console(config)# logging buffered 100 informational
```

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 ? (Y/N)[N]
```

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
```

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]
```

aaa logging

Use the **aaa logging** Global Configuration mode command to enable logging AAA logins. Use the **no** form of this command to disable logging AAA logins.

Syntax

```
aaa logging {login}
```

```
no aaa logging {login}
```

Parameters

login—Enables logging messages related to successful AAA login events, unsuccessful AAA login events and other AAA login-related events.

Default Configuration

Enabled.

Command Mode

Global Configuration mode

User Guidelines

This command enables logging messages related to successful login events, unsuccessful login events and other login-related events. Other types of AAA events are not subject to this command.

Example

The following example enables logging AAA login events.

```
console(config)# aaa logging login
```

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
```

management logging

Use the **management logging** Global Configuration mode command to enable logging Management Access List (ACL) deny events (rejected logins). Use the **no** form of this command to disable logging management access list events.

Syntax

management logging { *deny* }

no management logging { *deny* }

Parameters

deny—Enables logging messages related to management ACL deny actions (rejected logins).

Default Configuration

Logging management ACL deny events is enabled.

Command Mode

Global Configuration mode

User Guidelines

Other management ACL events are not subject to this command.

Example

The following example enables logging messages related to management ACL deny actions.

```
console(config)# management logging deny
```

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
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
```

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
```

```
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 bgEgGnt9
z6NHgZwKI5xKqF7cBtdllxmFgSEWuDhho5UedydAjVkkS5XR2... 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#
```

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	

```
OOB host Configuration
```

```
-----
```

IP address	Port	Facility	Severity	Description
2.1.1.200	514	local7	warning	

Remote Network Monitoring (RMON) Commands

show rmon statistics

Use the `show rmon statistics` Privileged 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

Privileged EXEC mode

Example

The following example displays RMON Ethernet statistics for port te0/1.

```
console# show rmon statistics te0/1
Port te0/1
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
Dropped	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.
Octets	Total number of octets of data (including those in bad packets) received on the network (excluding framing bits but including FCS octets).
Packets	Total number of packets (including bad packets, broadcast packets, and multicast packets) received.
Broadcast	Total number of good packets received and directed to the broadcast address. This does not include multicast packets.
Multicast	Total number of good packets received and directed to a multicast address. This number does not include packets directed to the broadcast address.
CRC Align Errors	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).
Collisions	Best estimate of the total number of collisions on this Ethernet segment.
Undersize Pkts	Total number of packets received, less than 64 octets long (excluding framing bits, but including FCS octets) and otherwise well formed.
Oversize Pkts	Total number of packets received, longer than 1518 octets (excluding framing bits, but including FCS octets) and otherwise well formed.
Fragments	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).

Field	Description
Jabbers	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).
64 Octets	Total number of packets (including bad packets) received that are 64 octets in length (excluding framing bits but including FCS octets).
65 to 127 Octets	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).
128 to 255 Octets	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).
256 to 511 Octets	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).
512 to 1023 Octets	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).
1024 to max	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).

rmon collection stats

Use the **rmon collection stats** Interface Configuration mode command to enable RMON MIB collecting history statistics (in groups) 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 mode.

show rmon collection stats

Use the `show rmon collection stats` Privileged 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

Privileged EXEC mode

Example

The following example displays all RMON history group statistics.

```
console# show rmon collection stats
```

Index	Interface	Interval	Requested Samples	Granted Samples	Owner
1	te0/1	30	50	50	CLI
2	te0/1	1800	50	50	Manager

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

Field	Description
Index	An index that uniquely identifies the entry.
Interface	The sampled Ethernet interface.
Interval	The interval in seconds between samples.
Requested Samples	The requested number of samples to be saved.
Granted Samples	The granted number of samples to be saved.
Owner	The entity that configured this entry.

show rmon history

Use the `show rmon history` Privileged 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

Privileged 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: te0/1      Interval: 1800
Requested samples: 50  Granted samples: 50
```

```
Maximum table size: 500
```

Time	Octets	Packets	Broadcast	Multicast	Util
Jan 18 2005 21:57:00	303595962 287696304	357568 275686	3289 2789	7287 5878	19% 20%
Jan 18 2005 21:57:30					

```
console# show rmon history 1 errors
```

```
Sample Set: 1          Owner: Me
Interface:te0/1      Interval: 1800
Requested samples: 50  Granted samples: 50
```

```
Maximum table size: 500 (800 after reset)
```

Time	CRC Align	Under size	Oversize	Fragments	Jabbers
Jan 18 2005 21:57:00	1 1	1 1	0 0	49 27	0 0
Jan 18 2005 21:57:30					

```

console# show rmon history 1 other

Sample Set: 1          Owner: Me
Interface: te0/1      Interval: 1800
Requested samples: 50  Granted samples: 50

Maximum table size: 500

Time                Droppe  Collisions
-----            d
Jan 18 2005 21:57:00  -----  0
Jan 18 2005 21:57:30    3        0
                        3

```

The following table describes significant fields shown in the display:

Field	Description
Time	Date and Time the entry is recorded.
Octets	Total number of octets of data (including those in bad packets and excluding framing bits but including FCS octets) received on the network.
Packets	Number of packets (including bad packets) received during this sampling interval.
Broadcast	Number of good packets received during this sampling interval that were directed to the broadcast address.
Multicast	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.
Utilization	Best estimate of the mean physical layer network utilization on this interface during this sampling interval, in hundredths of a percent.
CRC Align	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).

Field	Description
Undersize	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.
Oversize	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.
Fragments	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.
Jabbers	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).
Dropped	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.
Collisions	Best estimate of the total number of collisions on this Ethernet segment during this sampling interval.

rmon alarm

Use the **rmon alarm** Global Configuration modecommand to configure alarm conditions. Use the **no** form of this command to remove an alarm.

Syntax

```
rmon alarm index mib-object-id interval rising-threshold falling-threshold
rising-event falling-event [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)
- **rising-threshold**—Specifies the rising threshold value. (Range: 0–4294967295)
- **falling-threshold**—Specifies the falling threshold value. (Range: 0–4294967295)
- **rising-event**—Specifies the index of the event triggered when a rising threshold is crossed. (Range: 0–65535)
- **falling-event**—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 **rising-threshold**, 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 **rising-threshold**, a single rising alarm is generated. If the first sample (after this entry becomes valid) is less than or equal to **falling-threshold**, a single falling alarm is generated.

- **falling**—Specifies that if the first sample (after this entry becomes valid) is less than or equal to **falling-threshold**, 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
```

show rmon alarm-table

Use the **show rmon alarm-table** Privileged EXEC mode command to display a summary of the alarms table.

Syntax

```
show rmon alarm-table
```

Command Mode

Privileged 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
Index	An index that uniquely identifies the entry.
OID	Monitored variable OID.
Owner	The entity that configured this entry.

show rmon alarm

Use the `show rmon alarm` Privileged EXEC mode command to display alarm configuration.

Syntax

`show rmon alarm number`

Parameters

`alarm number`—Specifies the alarm index. (Range: 1–65535)

Command Mode

Privileged 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
Alarm	Alarm index.
OID	Monitored variable OID.
Last Sample Value	Value of the statistic during the last sampling period. For example, if the sample type is delta , this value is the difference between the samples at the beginning and end of the period. If the sample type is absolute , this value is the sampled value at the end of the period.
Interval	Interval in seconds over which the data is sampled and compared with the rising and falling thresholds.
Sample Type	Method of sampling the variable and calculating the value compared against the thresholds. If the value is absolute , the variable value is compared directly with the thresholds at the end of the sampling interval. If the value is delta , the variable value at the last sample is subtracted from the current value, and the difference is compared with the thresholds.
Startup Alarm	Alarm that is 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.

Field	Description
Rising Threshold	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.
Falling Threshold	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.
Rising Event	Event index used when a rising threshold is crossed.
Falling Event	Event index used when a falling threshold is crossed.
Owner	Entity that configured this entry.

rmon event

Use the **rmon event** Global Configuration modecommand 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 (password) used when an SNMP trap is sent. (Octet string; length: 0–127 characters). Note this must be a community used in the definition of an SNMP host using the “snmp-server host” command.
- **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
```

show rmon events

Use the `show rmon events` Privileged EXEC mode command to display the RMON event table.

Syntax

```
show rmon events
```

Command Mode

Privileged 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 23:58:1
2	High Broadcast	Log Trap		Manager	Jan 18 2006 23:59:4

The following table describes significant fields shown in the display:

Field	Description
Index	Unique index that identifies this event.
Description	Comment describing this event.
Type	Type of notification that the device generates about this event. Can have the following values: none , log , trap , log-trap . 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.
Community	If an SNMP trap is to be sent, it is sent with the SNMP community string specified by this octet string.
Owner	The entity that configured this event.
Last time sent	The time this entry last generated an event. If this entry has not generated any events, this value is zero.

show rmon log

Use the `show rmon log` Privileged 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

Privileged 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
```

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 size.

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
```


802.1X Commands

aaa authentication dot1x

Use the **aaa authentication dot1x** Global Configuration mode command to specify which servers are used for authentication when 802.1X authentication is enabled. Use the **no** form of this command to restore the default configuration.

Syntax

```
aaa authentication dot1x default {radius | none | {radius | none}}
```

```
no aaa authentication dot1x default
```

Parameters

- **radius** - Uses the list of all RADIUS servers for authentication
- **none** - Uses no authentication

Default Configuration

RADIUS server.

Command Mode

Global Configuration mode

User Guidelines

You can select either authentication by a RADIUS server, no authentication (**none**), or both methods.

If you require that authentication succeeds even if no RADIUS server response was received, specify **none** as the final method in the command line.

Example

The following example sets the 802.1X authentication mode to RADIUS server authentication. Even if no response was received, authentication succeeds.

```
console(config)# aaa authentication dot1x default radius none
```

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*—Specify an Ethernet port ID.

Default Configuration

Statistics on all ports are cleared.

Command Mode

Privileged EXEC mode

User Guidelines

This command clears all the counters displayed in the `show dot1x` and `show dot1x statistics` command.

Example

```
console# clear dot1x statistics
```

dot1x auth-not-req

Use the `dot1x auth-not-req` Interface Configuration (VLAN) mode command to enable unauthorized devices access to a VLAN. Use the `no` form of this command to disable access to a VLAN.

Syntax

```
dot1x auth-not-req
```

```
no dot1x auth-not-req
```

Parameters

N/A

Default Configuration

Access is enabled.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

A VLAN cannot be defined as an unauthenticated VLAN if it is an access VLAN or it is the native VLAN for some ports.

If a VLAN is configured as an unauthenticated VLAN, traffic tagged with that VLAN and received from a member port of that VLAN will be bridged regardless of whether the port/host is authorized or not.

The guest VLAN cannot be configured as unauthorized VLAN.

Example

The following example enables unauthorized devices access to VLAN 5.

```
console(config)# interface vlan 5  
console(config-if)# dot1x auth-not-req
```

dot1x authentication

Use the **dot1x authentication** Interface Configuration mode command to enable authentication methods on a port. Use the **no** format of the command to return to the default.

Syntax

```
dot1x authentication [802.1x]
```

```
no dot1x authentication
```

Parameters

- **802.1x**—Enables authentication based on 802.1X (802.1X-based authentication).

Default Configuration

802.1X-Based authentication is enabled.

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

Example

The following example enables authentication based on 802.1x and the station's MAC address on port te0/1:

```
console(config)# interface te0/1  
console(config-if)# dot1x authentication 802.1x
```

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
```

Parameters

N/A

Default Configuration

No VLAN is defined as a guest VLAN.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

Use the `dot1x guest-vlan enable` command to enable unauthorized users on an interface to access the guest VLAN.

A device can have only one global guest VLAN.

The guest VLAN must be a static VLAN and it cannot be removed.

The Default VLAN cannot be configured as guest VLAN.

An unauthorized VLAN cannot be configured as guest VLAN.

The guest VLAN cannot be configured on a monitoring port.

Example

The following example defines VLAN 2 as a guest VLAN.

```
console(config)# interface vlan 2  
console(config-if)# dot1x guest-vlan
```

dot1x guest-vlan enable

Use the `dot1x guest-vlan enable` Interface Configuration mode command to enable unauthorized users on the access interface to the guest VLAN. Use the `no` form of this command to disable access.

Syntax

`dot1x guest-vlan enable`

`no dot1x guest-vlan enable`

Parameters

N/A

Default Configuration

The default configuration is disabled.

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

The port cannot belong to the guest VLAN.

This command cannot be configured if the monitoring VLAN is enabled on the interface.

The port is added to the guest VLAN as an egress untagged port.

If the authentication mode is single-host or multi-host, the value of PVID is set to the guest VLAN_ID.

If the authentication mode is multi-sessions mode, the PVID is not changed and all untagged traffic and tagged traffic not belonging to the unauthenticated VLANs from unauthorized hosts are mapped to the guest VLAN.

If 802.1X is disabled, the port static configuration is reset.

See the User Guidelines of the [dot1x host-mode](#) command for more information.

Example

The following example enables unauthorized users on te0/1 to access the guest VLAN.

```
console(config)# interface te0/1
console(config-if)# dot1x guest-vlan enable
```

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 a 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
```

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 (Ethernet) Configuration mode

User Guidelines

Single-Host Mode

The single-host mode manages the authentication status of the port: the port is authorized if there is an authorized host. In this mode, only a single host can be authorized on the port.

When a port is unauthorized and the guest VLAN is enabled, untagged traffic is remapped to the guest VLAN. Tagged traffic is dropped unless the VLAN tag is the guest VLAN or the unauthenticated VLANs. If guest VLAN is not enabled on the port, only tagged traffic belonging to the unauthenticated VLANs is bridged.

When a port is authorized, untagged and tagged traffic from the authorized host is bridged based on the static VLAN membership configured at the port. Traffic from other hosts is dropped.

A user can specify that untagged traffic from the authorized host will be remapped to a VLAN that is assigned by a RADIUS server during the authentication process. In this case, tagged traffic is dropped unless the

VLAN tag is the RADIUS-assigned VLAN or the unauthenticated VLANs. See the [dot1x radius-attributes vlan](#) command to enable RADIUS VLAN assignment at a port.

The switch removes from the FDB all MAC addresses learned on a port when its authentication status is changed from authorized to unauthorized.

Multi-Host Mode

The multi-host mode manages the authentication status of the port: the port is authorized after at least one host is authorized.

When a port is unauthorized and the guest VLAN is enabled, untagged traffic is remapped to the guest VLAN. Tagged traffic is dropped unless the VLAN tag is the guest VLAN or the unauthenticated VLANs. If guest VLAN is not enabled on the port, only tagged traffic belonging to the unauthenticated VLANs is bridged.

When a port is authorized, untagged and tagged traffic from all hosts connected to the port is bridged, based on the static VLAN membership configured at the port.

A user can specify that untagged traffic from the authorized port will be remapped to a VLAN that is assigned by a RADIUS server during the authentication process. In this case, tagged traffic is dropped unless the VLAN tag is the RADIUS-assigned VLAN or unauthenticated VLANs. See the [dot1x radius-attributes vlan](#) command to enable RADIUS VLAN assignment at a port.

The switch removes from FDB all MAC addresses learned on a port when its authentication status is changed from authorized to unauthorized.

Multi-Sessions Mode

Unlike the single-host and multi-host modes (port-based modes) the multi-sessions mode manages the authentication status for each host connected to the port (session-based mode). If the multi-sessions mode is configured on a port, the port does not have any authentication status. Any number of hosts can be authorized on the port. The [dot1x max-hosts](#) command can limit the maximum number of authorized hosts allowed on the port.

Each authorized client requires a TCAM rule. If there is no available space in the TCAM, the authentication is rejected.

When using this command to change the port mode to **single-host** or **multi-host** when authentication is enabled, the port state is set to unauthorized.

If this command changes the port mode to **multi-session** when authentication is enabled, the state of all attached hosts is set to unauthorized.

To change the port mode to single-host or multi-host, set the port ([dot1x port-control](#)) to force-unauthorized, change the port mode to single-host or multi-host, and set the port to authorization auto.

Multi-sessions mode cannot be configured on the same interface together with policy-based VLANs configured by [switchport general map protocols-group vlan](#)

Tagged traffic belonging to the unauthenticated VLANs is always bridged regardless of whether a host is authorized or not.

When the guest VLAN is enabled, untagged and tagged traffic from unauthorized hosts not belonging to the unauthenticated VLANs, is bridged via the guest VLAN.

Traffic from an authorized hosts is bridged in accordance with the port static configuration. A user can specify that untagged and tagged traffic from the authorized host not belonging to the unauthenticated VLANs will be remapped to a VLAN that is assigned by a RADIUS server during the authentication process. See the [dot1x radius-attributes vlan](#) command to enable RADIUS VLAN assignment at a port.

The switch does not remove from the FDB the host MAC address learned on the port when its authentication status is changed from authorized to unauthorized. The MAC address will be removed after the aging timeout expires.

Example

```
console(config)# interface te0/1
console(config-if)# dot1x host-mode multi-host
```

dot1x max-hosts

Use the **dot1x max-hosts** Interface Configuration command to configure the maximum number of authorized hosts allowed on an interface. Use the **no** format of the command to return to the default.

Syntax

```
dot1x max-hosts count
```

```
no dot1x max-hosts
```

Parameters

- *count*—Specifies the maximum number of authorized hosts allowed on the interface. May be any 32-bits positive number.

Default Configuration

No limitation.

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

By default, the number of authorized hosts allowed on an interface is not limited. To limit the number of authorized hosts allowed on an interface, use the `dot1x max-hosts` command.

This command is relevant only for multi-session mode.

Example

The following example limits the maximum number of authorized hosts on Ethernet port te0/1 to 6:

```
console(config)# interface te0/1  
console(config-if)# dot1x max-hosts 6
```

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

- *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 (Ethernet) Configuration 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 te0/1
console(config-if)# dot1x max-req 6
```

dot1x port-control

Use the `dot1x port-control` Interface Configuration 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}`

Parameters

- **auto**—Enables 802.1X authentication on the port and causes it 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 sends and receives traffic without 802.1X-based client authentication.
- **force-unauthorized**—Denies all access through this port by forcing it 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 this port.

Default Configuration

The port is in the force-authorized state.

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

The switch removes all MAC addresses learned on a port when its authorization control is changed from **force-authorized** to another.



NOTE: It is recommended to disable spanning tree or to enable spanning-tree PortFast mode on 802.1X edge 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 sets 802.1X authentication on te0/1 to auto mode.

```
console(config)# interface te0/1
console(config-if)# dot1x port-control auto
```

dot1x radius-attributes filter-id

Use the `dot1x radius-attributes filter-id` Interface Configuration mode command to enable user-based ACL/Qos-Policy assignment. Use the `no` form of this command to disable user-based ACL/Qos-Policy assignment.

Syntax

```
dot1x radius-attributes filter-id
```

```
no dot1x radius-attributes filter-id
```

Parameters

N/A

Default Configuration

Disabled

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

User based ACL/Qos-Policy assignment is supported only in 802.1X multiple sessions.

The configuration of the parameter is allowed only when the port is Forced Authorized or Forced Unauthorized.

Example

The following example enables user-based ACL/Qos-Policy assignment.

```
console(config)# interface te0/1
console(config-if)# dot1x radius-attributes filter-id
```

dot1x radius-attributes vlan

Use the `dot1x radius-attributes vlan` Interface Configuration mode command to enable RADIUS-based VLAN assignment. Use the `no` form of this command to disable RADIUS-based VLAN assignment.

Syntax

```
dot1x radius-attributes vlan [reject | static]
```

```
no dot1x radius-attributes vlan
```

Parameters

- **reject**—If the RADIUS server authorized the supplicant, but did not provide a supplicant VLAN the supplicant is rejected. If the parameter is omitted, this option is applied by default.
- **static**—If the RADIUS server authorized the supplicant, but did not provide a supplicant VLAN, the supplicant is accepted.

Default Configuration

```
reject
```

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

If RADIUS provides invalid VLAN information, the authentication is rejected.

If a RADIUS server assigns a client with a non-existing VLAN, the switch creates the VLAN. The VLAN is removed when it is no longer being used.

If RADIUS provides valid VLAN information and the port does not belong to the VLAN received from RADIUS, it is added to the VLAN as an egress untagged port. When the last authorized client assigned to the VLAN becomes unauthorized or 802.1x is disabled on the port, the port is excluded from the VLAN.

If the authentication mode is single-host or multi-host, the value of PVID is set to the VLAN_ID.

If an authorized port, in single-host or multi-host mode, changes its status to unauthorized, the port static configuration is reset.

If the authentication mode is multi-sessions mode, the PVID is not changed and all untagged traffic and tagged traffic not belonging to the unauthenticated VLANs are mapped to the VLAN using TCAM.

If the last authorized host assigned to a VLAN received from RADIUS connected to a port in the multi-sessions mode changes its status to unauthorized, the port is removed from the VLAN, if it is not in the static configuration.

See the User Guidelines of the `dot1x host-mode` command for more information.

If 802.1X is disabled the port static configuration is reset.

If the `reject` keyword is configured and the RADIUS server authorizes the host but the RADIUS accept message does not assign a VLAN to the supplicant, authentication is rejected.

If the `static` keyword is configured and the RADIUS server authorizes the host then even though the RADIUS accept message does not assign a VLAN to the supplicant, authentication is accepted and the traffic from the host is bridged in accordance with port static configuration.

If this command is used when there are authorized ports/hosts, it takes effect at subsequent authentications. To manually re-authenticate, use the `dot1x re-authenticate` command.

The command cannot be configured on a port if it together with

- Multicast TV-VLAN
- Q-in-Q
- Voice VLAN

Example

Example 1. This example enables user-based VLAN assignment. If the RADIUS server authorized the supplicant, but did not provide a supplicant VLAN, the supplicant is rejected.

```
console(config)# interface te0/1
console(config-if)# dot1x radius-attributes vlan
console(config-if)# exit
```

Example 2. This example enables user-based VLAN assignment. If the RADIUS server authorized the supplicant but did not provide a supplicant VLAN, the supplicant is accepted and the static VLAN configurations is used.

```
console(config)# interface te0/1
console(config-if)# dot1x radius-attributes static
console(config-if)# exit
```

dot1x re-authenticate

The **dot1x re-authenticate** Privileged EXEC mode command manually initiates re-authentication of all 802.1X-enabled ports or the specified 802.1X-enabled port.

Syntax

```
dot1x re-authenticate [interface-id]
```

Parameters

- *interface-id*—Specifies an Ethernet port.

Default Configuration

If no port is specified, command is applied to all ports.

Command Mode

Privileged EXEC mode

Example

The following command manually initiates re-authentication of 802.1X-enabled te0/1:

```
console# dot1x re-authenticate te0/1
```

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 (Ethernet) Configuration mode

Example

```
console(config)# interface te0/1
console(config-if)# dot1x reauthentication
```

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`

Parameters

N/A

Default Configuration

Disabled.

Command Mode

Global Configuration mode

Example

The following example enables 802.1X globally.

```
console(config)# dot1x system-auth-control
```

dot1x timeout quiet-period

Use the `dot1x timeout quiet-period` Interface Configuration mode command to set the time interval that the device remains in a quiet state following a failed authentication exchange (for example, if 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 a client. (Range: 10–65535 seconds).

Default Configuration

The default quiet period is 60 seconds.

Command Mode

Interface (Ethernet) Configuration 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.

For 802.1x-based authentication, the number of failed logins is 1.

For 802.1x-based and MAC-based authentication methods, the quiet period is applied after each failed attempt.

Example

The following example sets the time interval that the device remains in the quiet state following a failed authentication exchange to 120 seconds.

```
console(config)# interface te0/1
console(config-if)# dot1x timeout quiet-period 120
```

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 (Ethernet) Configuration mode

User Guidelines

The command is only applied to the 802.1x authentication method.

Example

```
console(config)# interface te0/1
console(config-if)# dot1x timeout reauth-period 5000
```

dot1x timeout server-timeout

Use the `dot1x timeout server-timeout` Interface Configuration 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 (Ethernet) Configuration mode

User Guidelines

The actual timeout period can be determined by comparing the value specified by this command to the result of multiplying the number of retries

by the timeout period (both specified by the [radius-server retransmit](#) 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 te0/1
console(config-if)# dot1x timeout server-timeout 3600
```

dot1x timeout supp-timeout

Use the `dot1x timeout supp-timeout` Interface Configuration 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 (Ethernet) Configuration 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.

The command is only applied to the 802.1x authentication method.

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 te0/1
console(config-if)# dot1x timeout supp-timeout 3600
```

dot1x timeout tx-period

Use the `dot1x timeout tx-period` Interface Configuration 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: 30–65535 seconds).

Default Configuration

The default timeout period is 30 seconds.

Command Mode

Interface (Ethernet) Configuration 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.

The command is only applied to the 802.1x authentication method.

Example

The following command sets the time interval during which the device waits for a response to an EAP request/identity frame to 60 seconds.

```
console(config)# interface te0/1:  
console(config-if)# dot1x timeout tx-period 60
```

dot1x traps authentication failure

Use the `dot1x traps authentication failure` Global Configuration mode command to enable sending traps when an 802.1X authentication method failed. Use the `no` form of this command to return to the default.

Syntax

```
dot1x traps authentication failure [802.1x]
```

```
no dot1x traps authentication failure
```

Parameters

- `802.1x`—Enables traps for 802.1X-based authentication.

Default Configuration

All traps are disabled.

Command Mode

Global Configuration mode

User Guidelines

Any combination of the keywords are allowed. At least one keyword must be configured.

A rate limit is applied to the traps: not more than one trap of this type can be sent in 10 seconds.

Example

The following example enables sending traps when a MAC address fails to be authorized by the 802.1X mac-authentication access control.

```
console(config)# dot1x traps authentication failure 802.1x
```

dot1x traps authentication success

Use the `dot1x traps authentication success` Global Configuration mode command to enable sending traps when a host is successfully authorized by an 802.1X authentication method. Use the `no` form of this command to disable the traps.

Syntax

```
dot1x traps authentication success {[802.1x] [mac]}
```

```
no dot1x traps authentication success
```

Parameters

- `802.1x`—Enables traps for 802.1X-based authentication.
- `mac`—Enables traps for MAC-based authentication.

Default Configuration

Success traps are disabled.

Command Mode

Global Configuration mode

User Guidelines

Any combination of the keywords are allowed. At least one keyword must be configured.

A rate limit is applied to the traps: not more than one trap of this type can be sent in 10 seconds.

Example

The following example enables sending traps when a MAC address is successfully authorized by the 802.1X MAC-authentication access control.

```
console(config)# dot1x traps authentication success mac
```

dot1x violation-mode

Use the **dot1x violation-mode** Interface Configuration mode command to configure the action to be taken when an unauthorized host on authorized port in single-host mode attempts to access the interface. Use the **no** form of this command to return to default.

Syntax

```
dot1x violation-mode {restrict / protect / shutdown}
```

```
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 addresses are not learned.
- **protect**—Discard frames with source addresses that are not the supplicant address.
- **shutdown**—Discard frames with source addresses that are not the supplicant address and shutdown the port.

Default Configuration

Protect

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

The command is relevant only for single-host mode.

For BPDU messages whose MAC addresses are not the supplicant MAC address are not discarded in Protect mode.

BPDU message whose MAC addresses are not the supplicant MAC address cause a shutdown in Shutdown mode.

Example

```
console(config)# interface te0/1
console(config-if)# dot1x violation-mode protect
```

show dot1x

Use the **show dot1x** Privileged EXEC mode command to display the 802.1X interfaces or specified interface status.

Syntax

show dot1x [*interface interface-id* / *detailed*]

Parameters

- *interface-id*—Specify an Ethernet port.
- **detailed**—Displays information for non-present ports in addition to present ports.

Default Configuration

Display for all ports. If **detailed** is not used, only present ports are displayed.

Command Mode

Privileged EXEC mode

Examples

The following example displays authentication information for all interfaces of the switch supporting the full multi-sessions mode:

```
console# show dot1x
Authentication is enabled
Authenticating Servers: Radius, None
```

```
Unauthenticated VLANs: 100, 1000, 1021
Guest VLAN: VLAN 11, timeout 30 sec
Authentication failure traps are enables for 802.1x+mac
Authentication success traps are enables for 802.1x
te0/2
  Host mode: single-host
  Authentication methods: 802.1x+mac
  Port Adminstrated status: auto
  Port Operational status: authorized
  Guest VLAN: disabled
  VLAN Radius Attribute: enabled
  Server-timeout: 30 sec
  Aplied Authenticating Server: Radius
  Applied Authentication method: 802.1x
  Session Time (HH:MM:SS): 00:25:22
  MAC Address: 00:08:78:32:98:66
  Username: Bob
  Violation:
    Mode: restrict
    Trap: enabled
    Trap Min Interval: 20 sec
    Violations were detected: 9
  Reauthentication is enabled
  Reauthentication period: 3600 sec
  Silence period: 1800 sec
  Quiet Period: 60 sec
  Interfaces 802.1X-Based Parameters
    Tx period: 30 sec
    Supplicant timeout: 30 sec
    max-req: 2
  Authentication success: 2
  Authentication fails: 0
te0/3
```

```
Host mode: multi-host
Authentication methods: 802.1x+mac
Port Adminstrated status: auto
Port Operational status: authorized
Guest VLAN: disabled
VLAN Radius Attribute: disabled
Server-timeout: 30 sec
Aplied Authenticating Server: Radius
Applied Authentication method: 802.1x
Session Time (HH:MM:SS): 00:25:22
MAC Address: 00:08:78:32:98:66
Username: Bob
Violation:
  Mode: restrict
  Trap: enabled
  Trap Min Interval: 20 sec
  Violations were detected: 0
Reauthentication is enabled
Reauthentication period: 3600 sec
Silence period: 1800 sec
Quiet Period: 60 sec
Interfaces 802.1X-Based Parameters
  Tx period: 30 sec
  Supplicant timeout: 30 sec
  max-req: 2
Authentication success: 20
Authentication fails: 0
```

The following describes the significant fields shown in the display:

- **Port**—The port interface-id.
- **Host mode**—The port authentication configured mode. Possible values are.
 - single-host
 - multi-host

- multi-sessions
- **Port Administrated status**—The port administration (configured) mode. Possible values: force-auth, force-unauth, auto.
- **Port Operational status**—The port operational (actual) mode. Possible values: authorized or unauthorized.
- **Username**—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 authorized successfully.
- **Quiet period**—Number of seconds that the device remains in the quiet state following a failed authentication exchange (for example, the client provided an invalid password).
- **Silence period**—Number of seconds that If an authorized client does not send traffic during the silence period specified by the command, the state of the client is changed to unauthorized.
- **Tx period**—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.
- **Max req**—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.
- **Supplicant timeout**—Number of seconds that the device waits for a response to an EAP-request frame from the client before resending the request.
- **Server timeout**—Number of seconds that the device waits for a response from the authentication server before resending the request.
- **Session Time**—Amount of time (HH:MM:SS) that the user is logged in.
- **MAC address**—Supplicant MAC address.
- **Authentication success**—Number of times the state machine received a Success message from the Authentication Server.
- **Authentication fails**—Number of times the state machine received a Failure message from the Authentication Server.

show dot1x statistics

Use the `show dot1x statistics` Privileged EXEC mode command to display 802.1X statistics for the specified port.

Syntax

`show dot1x statistics interface interface-id`

Parameters

- *interface-id*—Specify an Ethernet port.

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example displays 802.1X statistics for `te0/1`.

```
console# show dot1x statistics interface te0/1
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
EapolFramesRx	Number of valid EAPOL frames of any type that have been received by this Authenticator.
EapolFramesTx	Number of EAPOL frames of any type that have been transmitted by this Authenticator.
EapolStartFramesRx	Number of EAPOL Start frames that have been received by this Authenticator.
EapolLogoffFramesRx	Number of EAPOL Logoff frames that have been received by this Authenticator.
EapolRespIdFramesRx	Number of EAP Resp/Id frames that have been received by this Authenticator.
EapolRespFramesRx	Number of valid EAP Response frames (other than Resp/Id frames) that have been received by this Authenticator.
EapolReqIdFramesTx	Number of EAP Req/Id frames that have been transmitted by this Authenticator.
EapolReqFramesTx	Number of EAP Request frames (other than Req/Id frames) that have been transmitted by this Authenticator.
InvalidEapolFramesRx	Number of EAPOL frames that have been received by this Authenticator for which the frame type is not recognized.
EapLengthErrorFramesRx	Number of EAPOL frames that have been received by this Authenticator in which the Packet Body Length field is invalid.
LastEapolFrameVersion	Protocol version number carried in the most recently received EAPOL frame.
LastEapolFrameSource	Source MAC address carried in the most recently received EAPOL frame.

show dot1x users

Use the `show dot1x users` Privileged EXEC mode command to display active 802.1X authorized users for the device.

Syntax

show dot1x users [*username username*]

Parameters

- **username *username***—Specifies the supplicant username (Length: 1–160 characters).

Default Configuration

Display all users.

Command Mode

Privileged EXEC mode

Example

The following commands displays all 802.1x users:

```
console# show dot1x users
```

Port	Username	MAC Address	Auth Server	Session Time	VLAN
te0/1	Bob	0008.3b71.1111	Remote	09:01:00	1020
te0/2	Allan	0008.3b79.8787	Remote	00:11:12	
te0/2	John	0008.3baa.0022	Remote	00:27:16	

Ethernet Configuration Commands



NOTE: Some CLI commands or parameters described below are not applicable for fiber ports. Refer to [Port Features](#).

interface

Use the **interface** Global Configuration mode command to enter Interface configuration mode in order to configure an interface.

Syntax

interface *interface-id*

Parameters

interface-id—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port, port-channel, VLAN, range, OOB, IP interface or tunnel.

Default Configuration

N/A

Command Mode

Global Configuration mode

Examples

Example 1—For Ethernet ports:

```
console(config)# interface te0/1
console(config-if)#
```

Example 2—For port channels (LAGs):

```
console(config)# interface po1
console(config-if)#
```

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, VLAN, or port-channel

Default Configuration

N/A

Command Mode

Interface (Ethernet, Port Channel, VLAN) Configuration mode

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 te0/1-4
console(config-if-range)#
```

shutdown

Use the **shutdown** Interface Configuration mode command to disable an interface. Use the **no** form of this command to restart a disabled interface.

Syntax

shutdown

no shutdown

Parameters

N/A

Default Configuration

The interface is enabled.

Command Mode

Interface Configuration mode

User Guidelines

The shutdown command set a value of ifAdminStatus (see RFC 2863) to DOWN. When ifAdminStatus is changed to DOWN, ifOperStatus will be also changed to DOWN.

The DOWN state of ifOperStatus means that the interface does not transmit/receive messages from/to higher levels. For example, if you shut down a VLAN, on which an IP interface is configured, bridging into the VLAN continues, but the switch cannot transmit and receive IP traffic on the VLAN.

If the switch shuts down an Ethernet port, it additionally shuts down the port MAC sublayer too.

If the switch shuts down a port channel, it additionally shuts down all ports of the port channel too.

Example

Example 1—The following example disables te0/4 operations.

```
console(config)# interface te0/4
console(config-if)# shutdown
console(config-if)#
```

Example 2—The following example restarts the disabled Ethernet port.

```
console(config)# interface te0/4
console(config-if)# no shutdown
```

```
console(config-if)#
```

Example 3—The following example shuts down vlan 100.

```
console(config)# interface vlan 100  
console(config-if)# shutdown  
console(config-if)#
```

Example 4—The following example shuts down tunnel 1.

```
console(config)# interface tunnel 1  
console(config-if)# shutdown  
console(config-if)#
```

Example 5—The following example shuts down Port Channel 3.

```
console(config)# interface po3  
console(config-if)# shutdown  
console(config-if)#
```

description

Use the **description** Interface (Ethernet, Port Channel) Configuration 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 (Ethernet, Port Channel) Configuration mode

Example

The following example adds the description 'SW#3' to te0/4.

```
console(config)# interface te0/4
console(config-if)# description SW#3
```

speed

Use the **speed** Interface (Ethernet, Port Channel) Configuration 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 / 10000}
```

```
no speed
```

Parameters

- 10—Forces 10 Mbps operation
- 100—Forces 100 Mbps operation
- 1000—Forces 1000 Mbps operation
- 10000—Forces 10000 Mbps operation

Default Configuration

The port operates at its maximum speed capability.

Command Mode

Interface (Ethernet, Port Channel) Configuration 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 te0/4 to 1000 Mbps operation.

```
console(config)# interface te0/4
console(config-if)# speed 1000
```

duplex

Use the **duplex** Interface (Ethernet, Port Channel) Configuration 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 (Ethernet, Port Channel) Configuration mode

Example

The following example configures gi0/1 to operate in full duplex mode.

```
console(config)# interface gi0/1
```

```
console(config-if)# duplex full
```

negotiation

Use the **negotiation** Interface (Ethernet, Port Channel) Configuration mode command to enable auto-negotiation operation for the speed and duplex parameters and master-slave mode of a given interface. Use the **no** form of this command to disable auto-negotiation.

Syntax

```
negotiation [capability [capability2... capability5]] [preferred {master | slave}]
```

```
no negotiation
```

Parameters

- **Capability**—Specifies the capabilities to advertise. (Possible values: 10h, 10f, 100h, 100f, 1000f).
 - 10h—Advertise 10 half-duplex
 - 10f—Advertise 10 full-duplex
 - 100h—Advertise 100 half-duplex
 - 100f—Advertise 100 full-duplex
 - 1000f—Advertise 1000 full-duplex
 - 10000—Advertise 10000 full-duplex
- **Preferred**—Specifies the master-slave preference:
 - **Master**—Advertise master preference
 - **Slave**—Advertise slave preference

Default Configuration

If capability is unspecified, defaults to list of all the capabilities of the port and preferred master mode.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

Example

The following example enables auto-negotiation on gi0/1.

```
console(config)# interface gi0/1
console(config-if)# negotiation
```

flowcontrol

Use the **flowcontrol** Interface (Ethernet, Port Channel) Configuration 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 of Flow Control.
- **on**—Enables Flow Control.
- **off**—Disables Flow Control.

Default Configuration

Flow control is Disabled.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

Use the **negotiation** command to enable **flow control auto**.

Flow Control is not supported on 10G fiber ports.

Example

The following example enables Flow Control on port gi0/1

```
console(config)# interface gi0/1
```

```
console(config-if)# flowcontrol on
```

mdix

Use the **mdix** Interface (Ethernet) Configuration 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 Auto.

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

This command is not supported on 10G fiber ports.

Example

The following example enables automatic crossover on port gi0/1.

```
console(config)# interface gi0/1  
console(config-if)# mdix auto
```

back-pressure

Use the **back-pressure** Interface (Ethernet) Configuration 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 (Ethernet) Configuration mode

User Guidelines

Back-pressure cannot be enabled when EEE is enabled.

This command is not supported on 10G fiber ports.

Example

The following example enables back pressure on port gi0/1.

```
console(config)# interface gi0/1
console(config-if)# back-pressure
```

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
```

clear counters

Use the **clear counters** Privileged EXEC mode command to clear counters on all or on a specific interface.

Syntax

```
clear 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.

Default Configuration

All counters are cleared.

Command Mode

Privileged EXEC mode

Example

The following example clears the statistics counters for te0/1.

```
console# clear counters te0/1
```

set interface active

Use the `set interface active` Privileged 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

Privileged 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 `te0/1`.

```
console# set interface active te0/1
```

show interfaces configuration

Use the `show interfaces configuration` Privileged EXEC mode command to display the configuration for all configured interfaces or for a specific interface.

Syntax

`show interfaces configuration [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.

Default Configuration

Display all interfaces. If detailed is not used, only present ports are displayed.

Command Mode

Privileged 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
te0/1	1G-Copper	Full	10000	Disabled	Off	Up	Disabled	Off
te0/2	1G-Copper	Full	1000	Disabled	Off	Up	Disabled	Off

PO	Type	Speed	Neg	Flow Control	Admin State
Po1			Disabled	Off	Up

show interfaces status

Use the `show interfaces status` Privileged EXEC mode command to display the status of all 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

Privileged EXEC mode

Default Configuration

Display for all interfaces. If detailed is not used, only present ports are displayed.

Example

The following example displays the status of all configured interfaces.

```

console# show interfaces status

```

Port	Type	Duplex	Speed	Neg	Flow ctrl	Link State	Back Pressure	Mdix Mode
te0/1	1G-Copper	Full	1000	Disabled	Off	Up	Disabled	Off
te0/2	1G-Copper	--	--	--	--	Down	--	--

PO	Type	Duplex	Speed	Neg	Flow control	Link State
Pol	1G	Full	10000	Disabled	Off	Up

show interfaces advertise

Use the `show interfaces advertise` Privileged EXEC mode command to display auto-negotiation advertisement information for all configured interfaces or for a specific interface.

Syntax

```
show interfaces advertise [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.

Default Configuration

Display for all interfaces. If detailed is not used, only present ports are displayed.

Command Mode

Privileged EXEC mode

Examples

The following examples display auto-negotiation information.

```
console# show interfaces advertise

Port    Type      Neg      Prefered  Operational Link
-----  -
te0/1   1G-      Enab     Master    -----
te0/2   Copper   le       Slave     1000f, 100f, 10f, 10h
        1G-      Enab     1000f
        Copper   le

console# show interfaces advertise te0/1
Port:te0/1
Type: 1G-Copper
Link state: Up
Auto Negotiation: enabled
Preference: Master
```

	10h	10f	100h	100	1000f
	---	---	----	f	-----
Admin Local link Advertisement	yes	yes	yes	---	yes
Oper Local link Advertisement	yes	yes	yes	-	yes
Remote Local link Advertisement	no	no	yes	yes	yes
Priority Resolution	-	-	-	yes	yes
				-	

```

console# show interfaces advertise te0/1
Port: te0/1
Type: 1G-Copper
Link state: Up
Auto negotiation: disabled.
Preference: Slave

```

show interfaces description

Use the `show interfaces description` Privileged EXEC mode command to display the description for all configured interfaces or for a specific interface.

Syntax

`show interfaces description [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.

Default Configuration

Display description for all interfaces. If `detailed` is not used, only present ports are displayed.

Command Mode

Privileged EXEC mode

Example

The following example displays the description of all configured interfaces.

```
console# show interfaces description

Port          Descriptions
te0/1         -----
te0/2         Port that should be used for management only
te0/3
te0/4

PO            Description
----         -
Po1           Output
```

show interfaces counters

Use the `show interfaces counters` Privileged 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.

Default Configuration

Display counters for all interfaces. If `detailed` is not used, only present ports are displayed.

Command Mode

Privileged EXEC mode

Example

The following example displays traffic seen by all the physical interfaces.

```
console# show interfaces counters te0/1
Port          InUcastPkts  InMcastPkts  InBcastPkts  InOctets
-----
te0/1         0            0            0            0
Port          OutUcastPkts OutMcastPkts OutBcastPkts OutOctets
-----
te0/1         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
InOctets	Number of received octets.
InUcastPkts	Number of received Unicast packets.
InMcastPkts	Number of received Unicast packets.

Field	Description
InBcastPkts	Number of received broadcast packets.
OutOctets	Number of transmitted octets.
OutUcastPkts	Number of transmitted Unicast packets.
OutMcastPkts	Number of transmitted Multicast packets.
OutBcastPkts	Number of transmitted Broadcast packets.
FCS Errors	Number of frames received that are an integral number of octets in length but do not pass the FCS check.
Single Collision Frames	Number of frames that are involved in a single collision, and are subsequently transmitted successfully.
Multiple Collision Frames	Number of frames that are involved in more than one collision and are subsequently transmitted successfully.
SQE Test Errors	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.
Deferred Transmissions	Number of frames for which the first transmission attempt is delayed because the medium is busy.
Late Collisions	Number of times that a collision is detected later than one slotTime into the transmission of a packet.
Excessive Collisions	Number of frames for which transmission fails due to excessive collisions.
Oversize Packets	Number of frames received that exceed the maximum permitted frame size.
Internal MAC Rx Errors	Number of frames for which reception fails due to an internal MAC sublayer receive error.
Received Pause Frames	Number of MAC Control frames received with an opcode indicating the PAUSE operation.

Field	Description
Transmitted Pause Frames	Number of MAC Control frames transmitted on this interface with an opcode indicating the PAUSE operation.

show ports jumbo-frame

Use the `show ports jumbo-frame` Privileged EXEC mode command to display the whether jumbo frames are enabled on the device.

Syntax

`show ports jumbo-frame`

Parameters

N/A

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example displays whether jumbo frames are enabled on the device.

```
console# show ports jumbo-frame
Jumbo frames are disabled
Jumbo frames will be enabled after reset
```

storm-control broadcast enable

Use the `storm-control broadcast enable` Interface Configuration mode command to enable storm control on a port. Use the `no` form of this command to disable storm control.

Syntax

storm-control broadcast enable

no storm-control broadcast enable

Parameters

N/A

Default Configuration

Disabled

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

Use the [storm-control include-multicast](#) Interface Configuration command to count Multicast packets and optionally unknown Unicast packets in the storm control calculation.

Example

```
console(config)# interface te0/1
console(config-if)# storm-control broadcast enable
```

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 kilobits per second of Broadcast traffic on a port. (Range 3500 – max port speed)

Default Configuration

8500

Command Mode

Interface (Ethernet) Configuration mode

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

Example 1—Set to specific level:

```
console(config)# interface te0/1
console(config-if)# storm-control broadcast level kbps 12345
```

storm-control include-multicast

Use the **storm-control include-multicast** Interface Configuration mode command to count Multicast packets in a 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

no storm-control include-multicast

Default Configuration

Disabled

Command Mode

Interface (Ethernet) Configuration mode

Example

```
console(config)# interface te0/1
console(config-if)# storm-control include-multicast
```

show storm-control

Use the `show storm-control` Privileged EXEC mode command to display the configuration of storm control for all ports or for a specific one.

Syntax

```
show storm-control [interface-id / detailed]
```

Parameters

- `interface-id`—Specifies an Ethernet port.
- `detailed`—Displays information for non-present ports in addition to present ports.

Default Configuration

Display for all interfaces. If `detailed` is not used, only present ports are displayed.

Command Mode

Privileged EXEC mode

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).

If the suppression level in percentage is translated to a rate that is lower than the minimum rate, the minimum rate is set.

Example

```
console# show storm-control
```

Port	State	Rate [Kbits/Sec]	Included
te0/1	Enabled	12345	Broadcast, Multicast, Unknown Unicast
te0/2	Disabled	100000	Broadcast

PHY Diagnostics Commands



NOTE: Some CLI commands or parameters described below are not applicable for fiber ports. Refer to [Port Features](#).

test cable-diagnostics tdr

Use the `test cable-diagnostics tdr` Privileged EXEC mode command to use Time Domain Reflectometry (TDR) technology to diagnose the quality and characteristics of a copper cable attached to a port.

Syntax

`test cable-diagnostics tdr interface interface-id`

Parameters

`interface-id`—Specifies an Ethernet port ID.

Command Mode

Privileged EXEC mode

User Guidelines

This command does not work on fiber ports (if they exist on the device). The port to be tested should be shut down during the test, unless it is a combination port with fiber port active. In this case, it does not need to be shut down, because the test does not work on fiber ports.

The maximum length of cable for the TDR test is 120 meters.

Example

Example 1 - Test the copper cables attached to port 1 (a copper port).

```
console# test cable-diagnostics tdr interface gi0/1
Cable is open at 64 meters
```

Example 2 - Test the copper cables attached to port 2 (a combo port with fiber active).

```
console# test cable-diagnostics tdr interface te0/2
Fiber ports are not supported
```

show cable-diagnostics tdr

Use the `show cable-diagnostics tdr` Privileged EXEC mode command to display information on the last Time Domain Reflectometry (TDR) test performed on all copper ports or on a specific copper port.

Syntax

```
show cable-diagnostics tdr [interface interface-id]
```

Parameters

- `interface-id`—Specify an Ethernet port ID.

Command Mode

Privileged EXEC mode

User Guidelines

The maximum length of cable for the TDR test is 120 meters.

Example

The following example displays information on the last TDR test performed on all copper ports.

```
console# show cable-diagnostics tdr

Port      Result      Length      Date
----      -
          [meters]
          -----

te0/1     OK
```

```
te0/2      Short      50                13:32:00 23 July 2010
te0/3      Test has not been performed
te0/4      Open       64                13:32:00 23 July 2010
```

show cable-diagnostics cable-length

Use the `show cable-diagnostics cable-length` Privileged 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 Ethernet port ID.

Command Mode

Privileged 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]
----          -
te0/1         < 50
te0/2         Copper not active
te0/3         110-140
```

show fiber-ports optical-transceiver

Use the `show fiber-ports optical-transceiver` Privileged EXEC mode command to display the optical transceiver diagnostics.

Syntax

`show fiber-ports optical-transceiver` [*interface interface-id* / *detailed*]

Parameters

- `interface-id`—Specify an Ethernet port ID.
- `detailed`—Displays information for non-present ports in addition to present ports.

Default Configuration

All ports are displayed. If `detailed` is not used, only present ports are displayed.

Command Mode

Privileged 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
-----
  te0/1    W      OK      OK      OK      OK      OK
  te0/2    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
```


EEE Commands



NOTE: Some CLI commands or parameters described below are not applicable for fiber ports. Refer to [Port Features](#).

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 as 1Giga, EEE always works regardless of whether the auto-negotiation status is enabled or disabled.

If auto-negotiation is not enabled on the port and its speed is less than 1 Giga, the EEE operational status is disabled.

Example

```
console(config)# eee enable
```

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 (Ethernet) Configuration mode

User Guidelines

If auto-negotiation is not enabled on the port and its speed is 1 Giga, the EEE operational status is disabled.

Example

```
console(config)# interface te0/1
console(config-if)# eee enable
```

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 (Ethernet) Configuration mode

User Guidelines

Enabling EEE LLDP advertisement enables devices to choose and change system wake-up times in order to get the optimal energy saving mode.

Example

```
console(config)# interface te0/1
console(config-if)# eee lldp enable
```

show eee

Use the `show eee EXEC` command to display EEE information.

Syntax

`show eee [interface-id]`

Parameters

`interface-id`—Specify an Ethernet port.

Defaults

N/A

Command Mode

Privileged EXEC mode

Examples

Example 1 - The following displays brief Information about all ports.

```
console# show eee
EEE globally enabled
EEE Administrative status is enabled on ports: te0/1-2, te0/4
EEE Operational status is enabled on ports: te0/1-2, te0/4
EEE LLDP Administrative status is enabled on ports: te0/1-3
EEE LLDP Operational status is enabled on ports: te0/1-2
```

Example 2 - The following is the information displayed when a port is in the Not Present state; no information is displayed if the port supports EEE.

```
console# show eee te0/1
Port Status: notPresent
EEE Administrative status: enabled
EEE LLDP Administrative status: enabled
```

Example 3 - The following is the information displayed when the port is in status DOWN.

```
console# show eee te0/1
Port Status: DOWN
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
    Speed 10G: EEE not supported
EEE Administrative status: enabled
EEE LLDP Administrative status: enabled
```

Example 4 - The following is the information displayed when the port is in status UP and does not support EEE.

```
console# show eee te0/2
Port Status: UP
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
    Speed 10G: EEE not supported
Current port speed: 1000Mbps
EEE Administrate status: enabled
EEE LLDP Administrate status: enabled
```

Example 5 - The following is the information displayed when the neighbor does not support EEE.

```
console# show eee te0/4
Port Status: UP
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
    Speed 10G: EEE not supported
Current port speed: 1000Mbps
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 - The following is the information displayed when EEE is disabled on the port.

```
console# show eee te0/1
Port Status: UP
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
    Speed 10G: EEE not supported
Current port speed: 1000Mbps
EEE Administrate status: disabled
EEE Operational status: disabled
EEE LLDP Administrate status: enabled
EEE LLDP Operational status: disabled
```

Example 7 - The following is the information displayed when EEE is running on the port, and EEE LLDP is disabled.

```
console# show eee te0/2
Port Status: UP
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
    Speed 10G: EEE not supported
Current port speed: 1000Mbps
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 - The following is the information displayed when EEE and EEE LLDP are running on the port.

```
console# show eee te0/3
Port Status: UP
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
    Speed 10G: EEE not supported
Current port speed: 1000Mbps
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 - The following is the information displayed when EEE is running on the port, EEE LLDP is enabled but not synchronized with the remote link partner.

```
console# show eee te0/4
Port Status: up
```

```
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
    Speed 10G: EEE not supported
Current port speed: 1000Mbps
EEE Remote status: enabled
EEE Administrate status: enabled
EEE Operational status: enabled
EEE LLDP Administrate 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 - The following is the information displayed when EEE and EEE LLDP are running on the port.

```
show eee te0/3
Port Status: UP
EEE capabilities:
    Speed 10M: EEE not supported
    Speed 100M: EEE supported
    Speed 1G: EEE supported
    Speed 10G: EEE not supported
Current port speed: 1000Mbps
EEE Remote status: enabled
EEE Administrate status: enabled
EEE Operational status: enabled
EEE LLDP Administrate 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

Green Ethernet



NOTE: Some CLI commands or parameters described below are not applicable for fiber ports. Refer to [Port Features](#).

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

Disabled.

Command Mode

Global Configuration mode

Example

```
console(config)# green-ethernet energy-detect
```

green-ethernet energy-detect (interface)

Use the `green-ethernet energy-detect` Interface configuration mode command to enable Green Ethernet-Energy-Detect mode on a port. Use the `no` form of this command, to disable it on a port.

Syntax

green-ethernet energy-detect
no green-ethernet energy-detect

Parameters

N/A

Default Configuration

Disabled.

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

Energy-Detect only works on copper ports. 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 te0/1  
console(config-if)# green-ethernet energy-detect
```

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
```

green-ethernet short-reach (interface)

Use the **green-ethernet short-reach** Interface Configuration mode command to enable green-ethernet short-reach mode on a port. Use the **no** form of this command to disable it on a port.

Syntax

green-ethernet short-reach

no green-ethernet short-reach

Parameters

N/A

Default Configuration

Disabled.

Command Mode

Interface (Ethernet) Configuration mode

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 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 te0/1
console(config-if)# green-ethernet short-reach
```

green-ethernet short-reach force

Use the **green-ethernet short-reach force** Interface Configuration mode command to force short-reach mode on a port. Use the **no** form of this command to return to default.

Syntax

```
green-ethernet short-reach force
no green-ethernet short-reach force
```

Parameters

N/A

Default Configuration

Short-reach mode is not forced.

Command Mode

Interface (Ethernet) Configuration mode

Example

```
console(config)# interface te0/1
console(config-if)# green-ethernet short-reach force
```

green-ethernet short-reach threshold

Use the **green-ethernet short-reach threshold** Global Configuration mode command to set the maximum cable length for applying short-reach. Use the **no** form of this command to return to default.

Syntax

green-ethernet short-reach threshold *cable-length*

no green-ethernet short-reach threshold

Parameters

cable-length—Specifies the maximum cable length (in meters) measured by VCT that allows applying short-reach mode (cable-length 0–70 meters)

Default Configuration

The default length is 40 meters.

Command Mode

Global Configuration mode

User Guidelines

Note that the automatic cable length measurement accuracy is ± 10 meters. i.e. a cable with a real length of 30 m may be evaluated in the range of 20m–40m. Length performance depends on the link partner signal quality, cable quality and whether the link partner also operates in short-reach mode.

Marvell recommends a default of 50m for any cable type.

However, Marvell tests show that the link partner can operate error free with a cable length of up to 80 m (cat 5e).

The user may choose to change the threshold parameter under certain circumstances.

Setting the threshold to 0 meters, basically results in the short reach feature always being disabled, because the threshold is always exceeded.

Example

```
console(config)# interface te0/1
console(config-if)# green-ethernet short-reach threshold 30
```

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`

Parameters

N/A

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

```
console# green-ethernet power-meter reset
```

show green-ethernet

Use the `show green-ethernet` Privileged EXEC mode command to display green-ethernet configuration and information.

Syntax

`show green-ethernet [interface-id / detailed]`

Parameters

- `interface-id`—Specifies an Ethernet port

- **detailed**—Displays information for non-present ports in addition to present ports.

Default Configuration

Display for all ports. If detailed is not used, only present ports are displayed.

Command Mode

Privileged EXEC mode

User Guidelines

The power savings displayed is relevant to the power saved by:

- Energy detect
- Short reach

The EEE power saving is dynamic by nature since it is based on port utilization and is therefore not taken into consideration.

The following describes the reasons for non-operation displayed by this command.

If there are a several reasons, then only the highest priority reason is displayed.

Example

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 exceeds threshold
6	LD	Port Link is Down – NA

```

console# show green-ethernet
Energy-Detect mode: Enabled
Short-Reach mode: Disabled
Power Savings: 24% (1.08W out of maximum 4.33W)

```

Cumulative Energy Saved: 33 [Watt*Hour]

Short-Reach cable length threshold: 50m

Port	Energy-Detect			Short-Reach			VCT Cable Length
	Admin	Oper	Reason	Admin	Force	Oper Reason	
te0/1	on	on		off	off	off	
te0/2	on	off	LU	on	off	off	< 50
te0/3	on	off	LU	off	off	off	

Port Channel Commands

channel-group

Use the **channel-group** Interface (Ethernet) Configuration 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**—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 (Ethernet) Configuration mode

Default mode is **on**.

Example

The following example forces port `te0/1` to join port-channel 1 without an LACP operation.

```
console(config)# interface te0/1
console(config-if)# channel-group 1 mode on
```

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 addresses.
- `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

Example

```
console(config)# port-channel load-balance src-dst-mac
```

show interfaces port-channel

Use the `show interfaces port-channel` Privileged 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

Privileged EXEC mode

Examples

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: 1,Inactive: te0/2-3
Po2      Active: 5 Inactive: te0/4
```

Address Table Commands

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

When this feature is enabled, unregistered Multicast traffic (as opposed to registered) will still be flooded.

All registered Multicast addresses will be forwarded to the Multicast groups. There are two ways to manage Multicast groups, one is the IGMP Snooping feature, and the other is the [bridge multicast forward-all](#) command.

Example

The following example enables bridge Multicast filtering.

```
console(config)# bridge multicast filtering
```

bridge multicast mode

Use the **bridge multicast mode** Interface (VLAN) Configuration 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* / *ipv4-group* / *ipv4-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 (VLAN) Configuration mode

User Guidelines

Use the **mac-group** option when using a network management system that uses a MIB based on the Multicast MAC address. Otherwise, it is recommended to use the **ipv4** 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	
mac-group	bridge multicast address	bridge multicast forbidden address
ipv4-group	bridge multicast ip-address	bridge multicast forbidden ip-address
ipv4-src-group	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
mac-group	MAC group address	MAC group address
ipv4-group	IP group address	IP group address
ipv4-src-group	(*)	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 an **mac-group** on VLAN 2.

```
console(config)# interface vlan 2
console(config-if)# bridge multicast mode mac-group
```

bridge multicast address

Use the **bridge multicast address** Interface (VLAN) Configuration 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 | ipv4-multicast-address}  
[[add | remove] {ethernet interface-list | port-channel port-channel-list}]
```

```
no bridge multicast address {mac-multicast-address}
```

Parameters

- **mac-multicast-address** | **ipv4-multicast-address**—Specifies the group 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 (VLAN) Configuration 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.

Examples

Example 1 - 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
```

Example 2 - 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 te0/1-2
```

bridge multicast forbidden address

Use the **bridge multicast forbidden address** Interface (VLAN) Configuration 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 | ipv4-multicast-address} {add | remove} {ethernet interface-list | port-channel port-channel-list}
```

```
no bridge multicast forbidden address {mac-multicast-address}
```

Parameters

- **mac-multicast-address** | **ipv4-multicast-address**—Specifies the group 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.

Default option is **add**.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

Before defining forbidden ports, the Multicast group should be registered, using [bridge multicast address](#).

You can execute the command before the VLAN is created.

Example

The following example forbids MAC address 0100.5e02.0203 on port te0/4 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 te0/4
```

bridge multicast ip-address

Use the **bridge multicast ip-address** Interface (VLAN) Configuration 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]* {*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.
- ***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 (VLAN) Configuration 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 specified 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  
te0/4
```

bridge multicast forbidden ip-address

Use the **bridge multicast forbidden ip-address** Interface (VLAN) Configuration 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 (VLAN) Configuration 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 te0/4 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 te0/4
```

bridge multicast source group

Use the **bridge multicast source group** Interface (VLAN) Configuration 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 (VLAN) Configuration 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 13.16.1.1 group
239.2.2.2
```

bridge multicast forbidden source group

Use the **bridge multicast forbidden source group** Interface (VLAN) Configuration 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 (VLAN) Configuration 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 port te0/4 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 te0/4
```

bridge multicast ipv6 mode

Use the **bridge multicast ipv6 mode** Interface (VLAN) Configuration 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 destination 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 (VLAN) Configuration 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	
mac-group	bridge multicast address	bridge multicast forbidden address
ipv6-group	bridge multicast ipv6 ip-address	bridge multicast ipv6 forbidden ip-address
ipv6-src-group	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
mac-group	MAC group address	MAC group address
ipv6-group	IPv6 group address	IPv6 group address
ipv6-src-group	(*)	IPv6 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 an **ip-group** on VLAN 2.

```

console(config)# interface vlan 2
console(config-if)# bridge multicast ipv6 mode
ip-group

```

bridge multicast ipv6 ip-address

Use the **bridge multicast ipv6 ip-address** Interface (VLAN) Configuration 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 (VLAN) Configuration 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

Example 1 - The following example registers the IPv6 address to the bridge table:

```
console(config)# interface vlan 8
console(config-if)# bridge multicast ipv6 ip-address
FF00:0:0:0:4:4:4:1
```

Example 2 - The following example registers the IPv6 address and adds ports statically.

```
console(config)# interface vlan 8
console(config-if)# bridge multicast ipv6 ip-address
FF00:0:0:0:4:4:4:1 add te0/1-2
```

bridge multicast ipv6 forbidden ip-address

Use the `bridge multicast ipv6 forbidden ip-address` Interface (VLAN) Configuration 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.

The default option is **add**.

Command Mode

Interface (VLAN) Configuration 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 te0/4 within VLAN 8.

```
console(config)# interface vlan 8
console(config-if)# bridge multicast ipv6 ip-address
FF00:0:0:0:4:4:4:1
console(config-if)# bridge multicast ipv6 forbidden ip-
address FF00:0:0:0:4:4:4:1 add te0/4
```

bridge multicast ipv6 source group

Use the **bridge multicast ipv6 source group** Interface (VLAN) Configuration 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.

The default option is **add**.

Command Mode

Interface (VLAN) Configuration 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 FF00:0:0:0:4:4:4:1
```

bridge multicast ipv6 forbidden source group

Use the **bridge multicast ipv6 forbidden source group** Interface (VLAN) Configuration mode command to forbid adding or removing a specific IPv6 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 (VLAN) Configuration 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 te0/4 on VLAN 8:

```
console(config)# interface vlan 8
console(config-if)# bridge multicast source 2001:0:0:0:4:4:4
group FF00:0:0:0:4:4:4:1
console(config-if)# bridge multicast forbidden source
2001:0:0:0:4:4:4:1 group FF00:0:0:0:4:4:4:1 add te0/4
```

bridge multicast unregistered

Use the **bridge multicast unregistered** Interface (Ethernet, Port Channel) Configuration mode command to configure forwarding 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 (Ethernet, Port Channel) Configuration 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 te0/1:

```
console(config)# interface te0/1
console(config-if)# bridge multicast unregistered filtering
```

bridge multicast forward-all

Use the **bridge multicast forward-all** Interface (VLAN) Configuration 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 (VLAN) Configuration mode

Example

The following example enables all Multicast packets on port te0/4 to be forwarded.

```
console(config)# interface vlan 2
console(config-if)# bridge multicast forward-all add te0/4
```

bridge multicast forbidden forward-all

Use the **bridge multicast forbidden forward-all** Interface (VLAN) Configuration 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.

The default option is **add**.

Command Mode

Interface (VLAN) Configuration 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 te0/1 within VLAN 2.

```
console(config)# interface vlan 2
console(config-if)# bridge multicast forbidden forward-all
add ethernet te0/1
```

mac address-table static

Use the **mac address-table static** Global Configuration mode command to add a 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**—MAC 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)
- **permanent**—The permanent static MAC address. The keyword is applied by the default.
- **delete-on-reset**—The delete-on-reset static MAC address.
- **delete-on-timeout**—The delete-on-timeout static MAC address.
- **secure**—The secure MAC address. May be used only in a secure mode.

Default Configuration

No static addresses are defined. The default mode for an added address is permanent.

Command Mode

Global Configuration mode

User Guidelines

Use the command to add a static MAC address with given time-to-live in any mode or to add a secure MAC address in a secure mode.

Each MAC address in the MAC address table is assigned two attributes: **type** and **time-to-live**.

The following value of time-of-live is supported:

- **permanent**—MAC address is saved until it is removed manually.
- **delete-on-reset**—MAC address is saved until the next reboot.
- **delete-on-timeout**—MAC address that may be removed by the aging timer.

The following types are supported:

- **static**—MAC address manually added by the command with the following keywords specifying its time-of-live:
 - **permanent**

- **delete-on-reset**
- **delete-on-timeout**

A static MAC address may be added in any port mode.

- **secure**— A MAC address added manually or learned in a secure mode. Use the **mac address-table static** command with the **secure** keyword to add a secure MAC address. The MAC address cannot be relearned.

A secure MAC address may be added only in a secure port mode.

- **dynamic**— a MAC address learned by the switch in non-secure mode. A value of its **time-to-live** attribute is **delete-on-timeout**.

Examples

Example 1 - The following example adds two permanent static MAC address:

```
console(config)# mac address-table static 00:3f:bd:45:5a:b1
vlan 1 interface te0/1
console(config)# mac address-table static 00:3f:bd:45:5a:b2
vlan 1 interface te0/1 permanent
```

Example 2 - The following example adds a deleted-on-reset static MAC address:

```
console(config)# mac address-table static 00:3f:bd:45:5a:b2
vlan 1 interface te0/1 delete-on-reset
```

Example 3 - The following example adds a deleted-on-timeout static MAC address:

```
console(config)# mac address-table static 00:3f:bd:45:5a:b2
vlan 1 interface te0/1 delete-on-timeout
```

Example 4 - The following example adds a secure MAC address:

```
console(config)# mac address-table static 00:3f:bd:45:5a:b2
vlan 1 interface te0/1 secure
```

clear mac address-table

Use the `clear mac address-table` Privileged EXEC mode command to remove learned or secure entries from the forwarding database (FDB).

Syntax

`clear mac address-table dynamic interface interface-id`

`clear mac address-table secure interface interface-id`

Parameters

- `dynamic interface interface-id`—Delete all dynamic (learned) addresses on the specified interface. The interface ID can be one of the following types: Ethernet port or port-channel. If interface ID is not supplied, all dynamic addresses are deleted.
- `secure interface interface-id`—Delete all the secure addresses learned on the specific interface. A secure address on a MAC address learned on ports on which port security is defined.

Default Configuration

For dynamic addresses, if `interface-id` is not supplied, all dynamic entries are deleted.

Command Mode

Privileged EXEC mode

Examples:

Example 1 - Delete all dynamic entries from the FDB.

```
console# clear mac address-table dynamic
```

Example 2 - Delete all secure entries from the FDB learned on secure port `te0/1`.

```
console# clear mac address-table secure interface te0/1
```

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
```

port security

Use the `port security` Interface (Ethernet, Port Channel) Configuration mode command to enable port security learning mode on an interface. Use the `no` form of this command to disable port security learning mode 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 by default.

The default mode is **discard**.

The default number of seconds is zero, but if **traps** is entered, a number of seconds must also be entered.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The command may be used only when the interface is in the regular (non-secure with unlimited MAC learning) mode.

See the [mac address-table static](#) command for information about MAC address attributes (type and time-to-live) definitions.

When the **port security** command enables the **lock** mode on a port all dynamic addresses learned on the port are changed to **permanent secure** addresses.

When the **port security** command enables a mode on a port differing from the **lock** mode all dynamic addresses learned on the port are deleted.

When the **no port security** command cancels a secure mode on a port all secure addresses defined on the port are changed to **dynamic** addresses.

Additionally to set a mode, use the **port security** command to set an action that the switch should perform on a frame which source MAC address cannot be learned.

Example

The following example forwards all packets to port te0/1 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)# interface te0/4
console(config-if)# port security mode lock
console(config-if)# port security forward trap 100
console(config-if)# exit
```

port security mode

Use the **port security mode** Interface (Ethernet, Port Channel) Configuration mode command configures the port security learning mode. Use the **no** form of this command to restore the default configuration.

Syntax

```
port security mode {max-addresses | lock }
no port security mode
```

Parameters

- **max-addresses**— Non-secure mode with limited learning dynamic MAC addresses. The static MAC addresses may be added on the port manually by the [mac address-table static](#) command.
- **lock**— Secure mode without MAC learning. The static and secure MAC addresses may be added on the port manually by the [mac address-table static](#) command.

Default Configuration

The default port security mode is

lock.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The default port mode is called regular. In this mode, the port allows unlimited learning of dynamic addresses.

The static MAC addresses may be added on the port manually by the `mac address-table static` command.

The command may be used only when the interface in the regular (non-secure with unlimited MAC learning) mode.

Use the `port security mode` command to change the default mode before the `port security` command.

Example

The following example sets the port security mode to

Lock for te0/4.

```
console(config)# interface te0/4
console(config-if)# port security mode lock
console(config-if)# port security
console(config-if)# exit
```

port security max

Use the `port security max` Interface (Ethernet, Port Channel) Configuration mode command to configure the maximum number of addresses that can be learned on the port while the port is in port, max-addresses or secure 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 (Ethernet, Port Channel) Configuration mode

User Guidelines

The command may be used only when the interface in the regular (non-secure with unlimited MAC learning) mode.

Use this command to change the default value before the [port security](#) command.

Example

The following example sets the port to limited learning mode:

```
console(config)# interface te0/4
console(config-if)# port security mode max
console(config-if)# port security max 20
console(config-if)# port security
console(config-if)# exit
```

port security routed secure-address

Use the `port security routed secure-address` Interface (Ethernet, Port Channel) Configuration mode command to add a MAC-layer secure address to a routed port. (port that has an IP address defined on it). Use the `no` form of this command to delete a MAC address from a routed port.

Syntax

```
port security routed secure-address mac-address
```

```
no port security routed secure-address mac-address
```

Parameters

`mac-address`—Specifies the MAC address.

Default Configuration

No addresses are defined.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode. It cannot be configured for a range of interfaces (range context).

User Guidelines

This command enables adding secure MAC addresses to a routed port in port security mode. The command is available when the port is a routed port and in port security mode. The address is deleted if the port exits the security mode or is not a routed port.

Example

The following example adds the MAC-layer address 00:66:66:66:66:66 to te0/1.

```
console(config)# interface te0/1
console(config-if)# port security routed secure-address
00:66:66:66:66:66
```

show mac address-table

Use the `show mac address-table` Privileged EXEC mode 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`—Displays entries for a specific VLAN.

- **interface-id**—Displays entries for a specific interface ID. The interface ID can be one of the following types: Ethernet port or port-channel.
- **mac-address**—Displays entries for a specific MAC address.

Default Configuration

If no parameters are entered, the entire table is displayed.

Command Mode

Privileged 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

Example 1 - Displays entire address table.

```
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	te0/1	static
1	00:a1:b0:69:63:f3	te0/2	dynamic
2	00:a1:b0:69:63:f3	te0/3	dynamic
te0/4	00:a1:b0:69:61:12	te0/4	dynamic

Example 2 - Displays address table entries containing the specified MAC address.

```
console# show mac address-table address 00:3f:bd:45:5a:b1
```

```
Aging time is 300 sec
```

VLAN	MAC Address	Port	Type
------	-------------	------	------

```
-----  
1          00:3f:bd:45:5a:b1    static    te0/4
```

show mac address-table count

Use the `show mac address-table count` Privileged 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

Privileged EXEC mode

Example

```
console# show mac address-table count  
This may take some time.  
Capacity : 16384  
Free     : 16379  
Used     : 5  
Secure   : 0  
Dynamic  : 2  
Static   : 2  
Internal : 1  
console#
```

show bridge multicast mode

Use the `show bridge multicast mode` Privileged 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

Privileged 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

show bridge multicast address-table

Use the `show bridge multicast address-table` Privileged EXEC mode command to display Multicast MAC addresses or IP Multicast address table information.

Syntax

`show bridge multicast address-table [vlan vlan-id]`

`show bridge multicast address-table [vlan vlan-id] [address mac-multicast-address] [format {ip | mac}]`

`show bridge multicast address-table [vlan vlan-id] [address ipv4-multicast-address] [source ipv4-source-address]`

`show bridge multicast address-table [vlan vlan-id] [address ipv6-multicast-address] [source ipv6-source-address]`

Parameters

- **vlan-id** *vlan-id*—Display entries for specified VLAN ID.
- **address**—Display entries for specified 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**—Applies if **mac-multicast-address** was selected. In this case either MAC or IP format can be displayed. Display entries for specified 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** —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** (only if **mac-multicast-address** was entered).

If VLAN ID is not entered, entries for all VLANs are displayed.

If MAC or IP address is not supplied, entries for all addresses are displayed.

Command Mode

Privileged 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 MAC groups.

Ports that were defined via the `bridge multicast forbidden forward-all` command are displayed in all forbidden MAC 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   te0/4

Multicast address table for VLANs in IPv4-GROUP bridging mode:
Vlan    MAC Address          Type          Ports
----  -
1       224.0.0.251         Dynamic       te0/2
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	te0/1

Forbidden ports for Multicast addresses:

Vlan	Group Address	Source Address	Ports
8	239.2.2.2	*	te0/4
8	239.2.2.2	1.1.1.11	te0/4

Multicast address table for VLANs in IPv6-GROUP bridging mode:

VLAN	IP/MAC Address	Type	Ports
8	ff02::4:4:4	Static	te0/1-2, te0/3, Po1

Forbidden ports for Multicast addresses:

VLAN	IP/MAC Address	Ports
8	ff02::4:4:4	te0/4

Multicast address table for VLANs in IPv6-SRC-GROUP bridging mode:

Vlan	Group Address	Source address	Type	Ports
8	ff02::4:4:4	*	Static	te0/1-2,te0/3,Po1
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	*	te0/4
8	ff02::4:4:4	fe80::200:7ff:f e00:200	te0/4

show bridge multicast address-table static

Use the `show bridge multicast address-table static` Privileged EXEC mode command to display the statically-configured Multicast addresses.

Syntax

```
show bridge multicast address-table static [vlan vlan-id] [all]
```

```
show bridge multicast address-table static [vlan vlan-id] [address mac-multicast-address] [mac] [ip]
```

```
show bridge multicast address-table static [vlan vlan-id] [address ipv4-multicast-address] [source ipv4-source-address]
```

```
show bridge multicast address-table static [vlan vlan-id] [address ipv6-multicast-address] [source ipv6-source-address]
```

Parameters

- `vlan vlan-id`—Specifies the VLAN ID.
- `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.
- `source`—Specifies the source address. The possible values are:
 - `ipv4-address`—Specifies the source IPv4 address.
 - `ipv6-address`—Specifies the source IPv6 address.

Default Configuration

When `all/mac/ip` is not specified, all entries (MAC and IP) will be displayed.

Command Mode

Privileged EXEC mode

User Guidelines

A MAC address can be displayed in IP format only if it is within the range 0100.5e00.0000— 0100.5e7f.ffff.

Example

The following example displays the statically-configured Multicast addresses.

```
console# show bridge multicast address-table static
MAC-GROUP table
Vlan          MAC Address          Ports
----          -
1             0100.9923.8787      te0/1, te0/2

Forbidden ports for multicast addresses:
Vlan          MAC Address          Ports
----          -
IPv4-GROUP Table
Vlan          IP Address           Ports
----          -
1             231.2.2.3           te0/1, te0/2
19           231.2.2.8           te0/2-3

Forbidden ports for multicast addresses:
Vlan          IP Address           Ports
----          -
1             231.2.2.3           te0/4
19           231.2.2.8           te0/3

IPv4-SRC-GROUP Table:
Vlan          Group Address        Source address        Ports
----          -
Forbidden ports for multicast addresses:
Vlan          Group Address        Source address        Ports
----          -
```

IPv6-GROUP Table

Vlan	IP Address	Ports
----	-----	-----
191	FF12::8	te0/1-4

Forbidden ports for multicast addresses:

Vlan	IP Address	Ports
----	-----	-----
11	FF12::3	te0/4
191	FF12::8	te0/4

IPv6-SRC-GROUP Table:

Vlan	Group Address	Source address	Ports
----	-----	-----	-----
192	FF12::8	FE80::201:C9A9:FE40:8988	te0/1-4

Forbidden ports for multicast addresses:

Vlan	Group Address	Source address	Ports
----	-----	-----	-----
192	FF12::3	FE80::201:C9A9:FE40:8988	te0/4

show bridge multicast filtering

Use the `show bridge multicast filtering` Privileged EXEC mode command to display the Multicast filtering configuration.

Syntax

`show bridge multicast filtering vlan-id`

Parameters

vlan-id—Specifies the VLAN ID. (Range: Valid VLAN)

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example displays the Multicast configuration for VLAN 1.

```
console# show bridge multicast filtering 1
Filtering: Enabled
VLAN: 1
Forward-All

Port          Static      Status
-----
te0/1         Forbidden  Filter
te0/2         Forward    Forward(s)
te0/3         -          Forward(d)
```

show bridge multicast unregistered

Use the **show bridge multicast unregistered** Privileged 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.

Default Configuration

Display for all interfaces.

Command Mode

Privileged EXEC mode

Example

The following example displays the unregistered Multicast configuration.

```
console# show bridge multicast unregistered
Port          Unregistered
-----
te0/1         Forward
te0/2         Filter
te0/3         Filter
```

show ports security

Use the `show ports security` Privileged EXEC mode command to display the port-lock status.

Syntax

`show ports security [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.

Default Configuration

Display for all interfaces. If `detailed` is not used, only present ports are displayed.

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      Maximum   Trap
Frequency
-----  -
te0/1   Enabled     Max-      Discard     3         Enabled
100
          Addresses
te0/2   Disabled    Max-      -           28        -
          Addresses
te0/3   Enabled     Lock      Discard     8         Disabled -
```

The following table describes the fields shown above.

Field	Description
Port	The port number.
Status	The port security status. The possible values are: Enabled or Disabled.
Action	The action taken on violation.
Maximum	The maximum number of addresses that can be associated on this port in the Max-Addresses mode.
Trap	The status of SNMP traps. The possible values are: Enable or Disable.
Frequency	The minimum time interval between consecutive traps.

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 / 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.

Default Configuration

Display for all interfaces. If detailed is not used, only present ports are displayed.

Command Mode

Privileged EXEC mode

Example

The following example displays dynamic addresses in all currently locked port:

Port	Status	Learning	Current	Maximum
te0/1	Disabled	Lock	0	10
te0/2	Disabled	Lock	0	1
te0/3	Disabled	Lock	0	1
te0/4	Disabled	Lock	0	1
...				

Port Monitor Commands

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*

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.
- *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 (Ethernet) Configuration mode

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.

Source ports cannot be destination ports at the same time.

The following restrictions apply to ports that are configured to be monitor ports:

- The port is not a member in a port-channel.
- An 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.

Note the following:

- 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.
 - An unknown unicast packet sent to VLAN 3 will egress from port 4 twice, once as normal forward and another time 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 is tagged and the other is not).
- When the port is configured to 802.1X auto mode, it will forward any mirrored traffic regardless of the 802.1X state. However, it will operate as a normal network port (forward traffic) only after authorization is done.
- Mirrored traffic is exposed to the 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 te0/2 to destination port te0/1.

```
console(config)# interface te0/1
console(config-if)# port monitor te0/2
```

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

show ports monitor

Use the **show ports monitor** EXEC mode command to display the port monitoring status.

Syntax

```
show ports monitor
```

Command Mode

User EXEC mode

Example

The following example displays the port monitoring status.

```
console# show ports monitor
```

Source port	Destination Port	Type	Status
-----	-----	-----	-----
te0/1	te0/4	RX,TX	Active
te0/2	te0/4	RX,TX	Active
te0/3	te0/4	RX	Active

sFlow Commands

sflow receiver

Use the `sflow receiver` Global Configuration mode command to define sFlow collector. Use the `no` form of this command to remove the definition of the collector.

Syntax

```
sflow receiver index {ipv4-address / ipv6-address / hostname} [port port]  
[max-datagram-size bytes]
```

```
no sflow receiver index
```

Parameters

- **index**—The index of the receiver. (Range: 1–8)
- **ipv4-address**—Pv4 address of the host to be used as an sFlow Collector.
- **ipv6-address**—IPv6 address of the host to be used as an sFlow Collector. 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 an sFlow Collector. Only translation to IPv4 addresses is supported.
- **port**—Port number for sflow messages. If unspecified, the port number defaults to 6343. The range is 1-65535.
- **bytes**—Specifies the maximum datagram size that can be sent. If unspecified, it defaults to 1400.

Default

No receiver is defined.

Command Mode

Global Configuration mode

User Guidelines

If the IP address of the sFlow receiver is set to 0.0.0.0, no sFlow datagrams are sent.

sflow flow-sampling

Use the `sflow flow-sampling` Interface Configuration mode command to enable sFlow Flow sampling and configure the average sampling rate of a specific port. Use the `no` form of this command to disable Flow sampling.

Syntax

```
sflow flow-sampling rate receiver-index [max-header-size bytes]
```

```
no sflow flow-sampling
```

Parameters

- `rate`—Specifies the average sampling rate (Range: 1, 1024–1073741823.)
- `receiver-index`—Index of the receiver/collector (Range: 1–8.)
- `bytes`—Specifies the maximum number of bytes that would be copied from the sampled packet. If unspecified, defaults to 128. (Range: 20–256.)

Default

Disabled

Command Mode

Interface Configuration mode

User Guidelines

A new sampling rate configuration is not immediately loaded to the hardware. It will be loaded to the hardware only after the next packet is sampled (based on the current sampling rate).

sflow counters-sampling

Use the `sflow counters-sampling` Interface Configuration mode command to enable sFlow Counters sampling and to configure the maximum interval of a

specific port. Use the **no** form of this command to disable sFlow Counters sampling.

Syntax

sflow counters-sampling *interval receiver-index*

no sflow counters-sampling

Parameters

- **interval**—Specifies the maximum number of seconds between successive samples of the interface counters. (Range: 1, 15–86400.)
- **receiver-index**—Index of the receiver/collector. (Range: 1–8.)

Default

Disabled

Command Mode

Interface Configuration mode

clear sflow statistics

Use the **clear sFlow statistics** Privileged EXEC mode command to clear sFlow statistics.

Syntax

clear sflow statistics [*interface-id*]

Parameters

interface-id—Specifies an interface ID. The interface ID must be an Ethernet port.

Command Mode

Privileged EXEC mode

User Guidelines

If no interface is specified by the user, the command clears all the sFlow statistics counters (including datagrams sent). If an interface is specified by the user, the command clears only the counter of the specific interface.

show sflow configuration

Use the `show sflow configuration` Privileged EXEC mode command to display the sFlow configuration for ports that are enabled for Flow sampling or Counters sampling.

Syntax

```
show sflow configuration [interface-id]
```

Parameters

interface-id—Specifies an interface ID. The interface ID must be an Ethernet port.

Command Mode

Privileged EXEC mode

Example

```
console# show sflow configuration
```

Receivers

Index	IP Address	Port	Max Datagram Size
1	0.0.0.0	6343	1400
2	172.16.1.2	6343	1400
3	0.0.0.0	6343	1400
4	0.0.0.0	6343	1400
5	0.0.0.0	6343	1400
6	0.0.0.0	6343	1400
7	0.0.0.0	6343	1400
8	0.0.0.0	6343	1400

Interfaces

Inter- Collector face	Flow Sampling	Counters Sampling	Max Header Size	Flow Collector	Counters Index	Counters Index
-----	-----	-----	-----	-----	-----	-----
te0/1	1/2048	60 sec	128	1		1
te0/2	1/4096	Disabled	128	0		2

show sflow statistics

Use the `show sflow statistics` Privileged EXEC mode command to display the sFlow statistics for ports that are enabled for Flow sampling or Counters sampling.

Syntax

`show sflow statistics [interface-id]`

Parameters

interface-id—Specifies an interface ID. The interface ID must be an Ethernet port.

Command Mode

Privileged EXEC mode

Example

```
console# show sflow statistics
Total sFlow datagrams sent to collectors: 100
```

Interface	Packets sampled	Datagrams sent to collector
-----	-----	-----
te0/1	30	50

te0/2	30	50
te0/3	30	50

LLDP Commands

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
```

lldp transmit

Use the `lldp transmit` Interface (Ethernet) 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 (Ethernet) Configuration mode

```
console(config-if)#
```

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 te0/1  
console(config-if)# lldp transmit
```

Ildp receive

Use the **lldp receive** Interface (Ethernet) Configuration mode command to enable receiving LLDP on an interface. Use the **no** form of this command to stop receiving LLDP on an Interface (Ethernet) Configuration mode interface.

Syntax

```
lldp receive
```

```
no lldp receive
```

Parameters

N/A

Default Configuration

Enabled

Command Mode

Interface (Ethernet) Configuration 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 te0/1
console(config-if)# lldp receive
```

Ildp 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
```

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
```

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
```

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 25% of the LLDP timer interval.

Example

The following example sets the LLDP transmission delay to 10 seconds.

```
console(config)# lldp tx-delay 10
```

lldp optional-tlv

Use the **lldp optional-tlv** Interface (Ethernet) Configuration mode command to specify which optional TLVs are transmitted. Use the **no** form of this command to restore the default configuration.

For 802.1, see the [lldp optional-tlv 802.1](#) command.

Syntax

`lldp optional-tlv tlv [tlv2 ... tlv5 | none]`

Parameters

- **tlv**—Specifies the TLVs 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**—Clear all optional TLVs from the interface.

If the 802.1 protocol is selected, see the command below.

Default Configuration

The following TLV are transmitted:

- sys-name
- sys-cap

Command Mode

Interface (Ethernet) Configuration mode

Example

The following example specifies that the port description TLV is transmitted on te0/2.

```
console(config)# interface te0/2
console(config-if)# lldp optional-tlv port-desc
```

lldp optional-tlv 802.1

Use the **lldp optional-tlv 802.1** Interface (Ethernet) Configuration mode command to specify whether to transmit the 802.1 TLV. Use the **no** form of this command to revert to the default setting.

Syntax

lldp optional-tlv 802.1 pvid *{enable / disable}*- The PVID is advertised or not advertised.

no lldp optional-tlv 802.1 pvid - The PVID advertise state is returned to default.

lldp optional-tlv 802.1 ppvid add *ppvid* - The Protocol Port VLAN ID (PPVID) is advertised. The PPVID is the PVID that is used depending on the packet's protocol.

lldp optional-tlv 802.1 ppvid remove *ppvid* - The PPVID is not advertised.

lldp optional-tlv 802.1 vlan add *vlan-id* - This *vlan-id* is advertised.

lldp optional-tlv 802.1 vlan remove *vlan-id* - This *vlan-id* is not advertised.

lldp optional-tlv 802.1 protocol add {*stp* / *rstp* / *mstp* / *pause* / *802.1x* / *lacp* / *gvrp*} - The protocols selected are advertised.

lldp optional-tlv 802.1 protocol remove {*stp* / *rstp* / *mstp* / *pause* / *802.1x* / *lacp* / *gvrp*} - The protocols selected are not advertised.

Parameters

- **lldp optional-tlv 802.1 pvid {*enable* / *disable*}**—Advertises or stop advertise the PVID of the port.
- **lldp optional-tlv 802.1 ppvid add/remove *ppvid***—Adds/removes PPVID for advertising. (range: 0–4094). PPVID = 0 indicates that the port is not capable of supporting port and protocol VLANs and/or the port is not enabled with any protocol VLANs.
- **add/remove *vlan-id***—Adds/removes VLAN for advertising (range: 0–4094).
- **add/remove {*stp* / *rstp* / *mstp* / *pause* / *802.1x* / *lacp* / *gvrp*}**—Add specifies to advertise the specified protocols; remove specifies not to advertise the specified protocol.

Default Configuration

The following 802.1 TLV is transmitted:

Command Mode

Interface (Ethernet) Configuration mode

Example

```
console(config)# lldp optional-tlv 802.1 protocol add stp
```

Ildp management-address

Use the `lldp management-address` Interface (Ethernet) Configuration 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**—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 (Ethernet) Configuration mode

User Guidelines

Each port can advertise one IP address.

Example

The following example sets the LLDP management address advertisement mode to **automatic** on te0/2.

```
console(config)# interface te0/2
console(config-if)# lldp management-address automatic
```

lldp notifications

Use the **lldp notifications** Interface (Ethernet) Configuration 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 (Ethernet) Configuration mode

Example

The following example enables sending LLDP notifications on te0/1.

```
console(config)# interface te0/1
console(config-if)# lldp notifications enable
```

Ildp 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
```

Ildp med

Use the `lldp med` Interface (Ethernet) Configuration 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... tlvA] | disable}`

`no lldp med`

Parameters

- `enable`—Enable LLDP MED

- **tlv**—Specifies the TLV that should be included. Available TLVs are: Network-Policy, Location, 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 (Ethernet) Configuration mode

Example

The following example enables LLDP MED with the **location** TLV on te0/3.

```
console(config)# interface te0/3
console(config-if)# lldp med enable location
```

Ildp med notifications topology-change

Use the **lldp med notifications topology-change** Interface (Ethernet) Configuration 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 (Ethernet) Configuration mode

Example

The following example enables sending LLDP MED topology change notifications on te0/2.

```
console(config)# interface te0/2
console(config-if)# lldp med notifications topology-change
enable
```

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 to 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
```

lldp med network-policy (global)

Use the `lldp med network-policy` Global Configuration mode command to define a LLDP MED network policy.

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-signal 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 te0/1
console(config-if)# lldp med network-policy add 1

```

lldp med network-policy (interface)

Use the **lldp med network-policy** Interface (Ethernet) Configuration 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 (Ethernet) Configuration 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 te0/1
console(config-if)# lldp med network-policy add 1
```

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 te0/1
```

Ildp med location

Use the `lldp med location` Interface (Ethernet) Configuration 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 (Ethernet) Configuration mode

Example

The following example configures the LLDP MED location information on te0/2 as a civic address.

```
console(config)# interface te0/2
console(config-if)# lldp med location civic-address
616263646566
```

lldp chassis-id

Use the **lldp chassis-id** Global Configuration mode command to configure the source of the chassis ID of the port. Use the **no** form of this command to restore the chassis ID source to default.

Syntax

```
lldp chassis-id {mac-address / host-name}
```

```
no lldp chassis-id
```

Parameters

- **mac-address**—Specifies the chassis ID to use the device MAC address.
- **host-name**—Specifies the chassis ID to use the device configured host name.

Default Configuration

MAC address.

Command Mode

Global Configuration mode

User Guidelines

The host name should be configured to be a unique value.

If the chassis ID configured to be used in LLDP packets is empty, LLDP uses the default chassis ID (specified above).

Example

The following example configures the chassis ID to be the MAC address.

```
console(config)# lldp chassis-id mac-address
```

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* / *detailed*]

Parameters

- **interface-id**—Specifies the port ID.
- **detailed**—Displays information for non-present ports in addition to present ports.

Default Configuration

Display for all ports. If `detailed` is not used, only present ports are displayed.

Command Mode

Privileged EXEC mode

Examples

Example 1 - Display LLDP configuration for all ports.

```
console# show lldp configuration
State: Enabled
Timer: 30 Seconds
Hold multiplier: 4
Reinit delay: 2 Seconds
Tx delay: 2 Seconds
Notifications interval: 5 seconds
Port      State  Optional TLVs      Address      Notifications
-----  -
te0/1     RX,TX PD, SN, SD, SC    172.16.1.1  Disabled
te0/2     TX     PD, SN             172.16.1.1  Disabled
te0/3     RX,TX PD, SN, SD, SC    None         Disabled
te0/4     RX,TX D, SN, SD, SC automatic     Disabled
```

Example 2 - Display LLDP configuration single port.

```
console# show lldp configuration te0/1
State: Enabled
LLDP state: Enabled
Timer: 30 Seconds
Hold multiplier: 4
Reinit delay: 2 Seconds
Tx delay: 2 Seconds
Notifications Interval: 5 Seconds
Port      State      Optional TLVs  Address Notifications
-----  -
te0/1     Rx and Tx  None           Disabled
802.3 optional TLVs: None
802.1 optional TLVs
PVID: Disabled
```


PPVIDs:

VLANs:

Protocols:

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

Field	Description
Timer	The time interval between LLDP updates.
Hold multiplier	The amount of time (as a multiple of the timer interval) that the receiving device holds a LLDP packet before discarding it.
Reinit timer	The minimum time interval an LLDP port waits before re-initializing an LLDP transmission.
Tx delay	The delay between successive LLDP frame transmissions initiated by value/status changes in the LLDP local systems MIB.
Port	The port number.
State	The port's LLDP state.
Optional TLVs	Optional TLVs that are advertised. Possible values are: PD - Port description SN - System name SD - System description SC - System capabilities
Address	The management address that is advertised.
Notifications	Indicates whether LLDP notifications are enabled or disabled.
PVID	Port VLAN ID advertised.
PPVID	Protocol Port VLAN ID advertised.

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* / *detailed*]

Parameters

- **interface-id**—Specifies the port ID.
- **detailed**—Displays information for non-present ports in addition to present ports.

Default Configuration

If no port ID is entered, the command displays information for all ports. If detailed is not used, only present ports are displayed.

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.
Network policy 1
-----
Application type: voiceSignaling
VLAN ID: 1 untagged
Layer 2 priority: 0
DSCP: 0
Port Capabilities Network Policy Location Notifications Inventory
-----
te0/1   Yes           Yes           Yes           Enabled       Yes
te0/2   Yes           Yes           No            Enabled       No
te0/3   No            No            No            Enabled       No
```

Example 2 - The following example displays the LLDP MED configuration for te0/1.

```
console# show lldp med configuration te0/1
```

Port	Capabilities	Network Policy	Location	Notifications	Inventory
te0/1	Yes	Yes	Yes	Enabled	Yes

Network policies:
Location:
Civic-address: 61:62:63:64:65:66

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

User 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

```

console# show lldp local tlvs-overloading te0/1
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

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 `te0/1` and `2`.

```
console# show lldp local te0/1
Device ID: 0060.704C.73FF
Port ID: te0/1
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 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
console# show lldp local te0/2
LLDP is disabled.
```

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]
```

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

User Guidelines

A TLV value that cannot be displayed as an ASCII string is displayed as a hexadecimal string.

Examples

Example 1 - The following example displays information about neighboring devices discovered using LLDP on all ports on which LLDP is enabled and who are up.

Location information, if it exists, is also displayed.

```
console# show lldp neighbors
```

System capability legend:

B - Bridge; R - Router; W - Wlan Access Point; T - telephone;

D - DOCSIS Cable Device; H - Host; r - Repeater;

TP - Two Ports MAC Relay; S - S-VLAN; C - C-VLAN; O - Other

Port Device ID Port ID System Name Capabilities TTL

te0/1	00:00:00:11:11:11	te0/1	ts-7800-2	B	90
te0/1	00:00:00:11:11:11	te0/1	ts-7800-2	B	90
te0/2	00:00:26:08:13:24	te0/3	ts-7900-1	B, R	90
te0/3	00:00:26:08:13:24	te0/2	ts-7900-2	W	90

Example 2 - The following example displays information about neighboring devices discovered using LLDP on port 1.

```
console# show lldp neighbors te0/1
Device ID: 00:00:00:11:11:11
Port ID: te0/1
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
Port	The port number.
Device ID	The neighbor device's configured ID (name) or MAC address.
Port ID	The neighbor device's port ID.
System name	The neighbor device's administratively assigned name.
Capabilities	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
System description	The neighbor device's system description.
Port description	The neighbor device's port description.
Management address	The neighbor device's management address.
Auto-negotiation support	The auto-negotiation support status on the port. (supported or not supported)
Auto-negotiation status	The active status of auto-negotiation on the port. (enabled or disabled)
Auto-negotiation Advertised Capabilities	The port speed/duplex/flow-control capabilities advertised by the auto-negotiation.

Field	Description
Operational MAU type	The port MAU type.
LLDP MED	
Capabilities	The sender's LLDP-MED capabilities.
Device type	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.
LLDP MED - Network Policy	
Application type	The primary function of the application defined for this network policy.
Flags	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.
VLAN ID	The VLAN identifier for the application.
Layer 2 priority	The Layer 2 priority used for the specified application.
DSCP	The DSCP value used for the specified application.
LLDP MED - Power Over Ethernet	
Power type	The device power type. The possible values are: Power Sourcing Entity (PSE) or Power Device (PD).
Power Source	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.
Power priority	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.

Field	Description
Power value	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.
LLDP MED - Location	
Coordinates, Civic address, ECS ELIN.	The location information raw data.

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 / detailed]`

Parameters

- `interface-id`—Specifies the port ID.
- `detailed`—Displays information for non-present ports in addition to present ports.

Default Configuration

If no port ID is entered, the command displays information for all ports. If `detailed` is not used, only present ports are displayed.

Command Mode

User EXEC mode

Example

```

console# 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

	TX Frames		RX Frame		RX	TLVs		RX
Ageouts								

Port	Total	Total	Discarded	Errors	Discarded	Unrecognized	
Total							

```

-----
-----
te0/1   730   850   0     0     0     0     0
te0/2   0     0     0     0     0     0     0
te0/3   730   0     0     0     0     0     0
te0/4   0     0     0     0     0     0     0

```

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

Field	Description
Port	The port number.
Device ID	The neighbor device's configured ID (name) or MAC address.
Port ID	The neighbor device's port ID.
System name	The neighbor device's administratively assigned name.
Capabilities	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
System description	The neighbor device's system description.
Port description	The neighbor device's port description.
Management address	The neighbor device's management address.

Field	Description
Auto-negotiation support	The auto-negotiation support status on the port. (Supported or Not Supported)
Auto-negotiation status	The active status of auto-negotiation on the port. (Enabled or Disabled)
Auto-negotiation Advertised Capabilities	The port speed/duplex/flow-control capabilities advertised by the auto-negotiation.
Operational MAU type	The port MAU type.
LLDP MED	
Capabilities	The sender's LLDP-MED capabilities.
Device type	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.
LLDP MED - Network Policy	
Application type	The primary function of the application defined for this network policy.
Flags	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.
VLAN ID	The VLAN identifier for the application.
Layer 2 priority	The Layer 2 priority used for the specified application.
DSCP	The DSCP value used for the specified application.
LLDP MED - Location	
Coordinates, Civic address, ECS ELIN.	The location information raw data.

Spanning-Tree Commands

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

Parameters

N/A

Default Configuration

Spanning-tree is enabled.

Command Mode

Global Configuration mode

Example

The following example enables spanning-tree functionality.

```
console(config)# spanning-tree
```

spanning-tree mode

Use the **spanning-tree mode** Global Configuration mode command to select which Spanning Tree Protocol (STP) protocol to run. 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 STP is enabled.
- **rstp**—Specifies that the Rapid STP is enabled.
- **mst**—Specifies that the Multiple STP is enabled.

Default Configuration

The default is STP.

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 enables MSTP.

```
console(config)# spanning-tree mode mst
```

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

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
```

spanning-tree hello-time

Use the `spanning-tree hello-time` Global Configuration mode command to configure 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

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
```

spanning-tree max-age

Use the **spanning-tree max-age** Global Configuration mode command to configure the STP 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 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
```

spanning-tree priority

Use the `spanning-tree priority` Global Configuration mode command to configure the device STP 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

Default priority = 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. When more than one switch has the lowest priority, the switch with the lowest MAC address is selected as the root.

Example

The following example configures the spanning-tree priority to 12288.

```
console(config)# spanning-tree priority 12288
```

spanning-tree disable

Use the **spanning-tree disable** Interface (Ethernet, Port Channel) Configuration 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

Parameters

N/A

Default Configuration

Spanning tree is enabled on all ports.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

Example

The following example disables the spanning tree on te0/4

```
console(config)# interface te0/4  
console(config-if)# spanning-tree disable
```

spanning-tree cost

Use the **spanning-tree cost** Interface (Ethernet, Port Channel) Configuration 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
TenGigabit Ethernet (10000 Mbps)	2000	2
Gigabit Ethernet (1000 Mbps)	20,000	4
Ethernet (10 Mbps)	2,000,000	100

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

Example

The following example configures the spanning-tree cost on te0/4 to 35000.

```
console(config)# interface te0/4
console(config-if)# spanning-tree cost 35000
```

spanning-tree port-priority

Use the **spanning-tree port-priority** Interface (Ethernet, Port Channel) Configuration 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 is 128.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The priority value must be a multiple of 16.

Example

The following example configures the spanning priority on te0/4 to 96

```
console(config)# interface te0/4
console(config-if)# spanning-tree port-priority 96
```

spanning-tree portfast

Use the **spanning-tree portfast** Interface (Ethernet, Port Channel) Configuration 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 Bridge Protocol Data Units (BPDUs) received on the interface) before putting the interface into the PortFast mode.

Default Configuration

PortFast mode is disabled.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

Example

The following example enables the PortFast mode on te0/4.

```
console(config)# interface te0/4
console(config-if)# spanning-tree portfast
```

spanning-tree link-type

Use the `spanning-tree link-type` Interface (Ethernet, Port Channel) Configuration mode command to override the default link-type setting determined by the port duplex mode, and enable 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 (Ethernet, Port Channel) Configuration mode

Example

The following example enables shared spanning-tree on te0/4.

```
console(config)# interface te0/4
console(config-if)# spanning-tree link-type shared
```

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: 1–200,000,000.

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 selected, the switch calculates the default cost as 100.
- If the long method is selected, the switch calculates the default cost as 20000.

Example

The following example sets the default path cost method to Long.

```
console(config)# spanning-tree pathcost method long
```

spanning-tree bpdud (Global)

Use the `spanning-tree bpdud` Global Configuration mode command to define Bridge Protocol Data Unit (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 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 all ports with the spanning tree disabled and BPDU handling mode of flooding. Tagged BPDU packets are filtered.

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.

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 bpdu flooding
```

spanning-tree bpdu (Interface)

Use the **spanning-tree bpdu** Interface (Ethernet, Port Channel) Configuration 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 bpdu {filtering / flooding}
```

```
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 ports with the spanning tree disabled and BPDU handling mode of flooding. Tagged BPDU packets are filtered.

Default Configuration

The `spanning-tree bpduglobal` command determines the default configuration.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

Example

The following example defines the BPDU packet as **flooding** when the spanning tree is disabled on te0/3.

```
console(config)# interface te0/3
console(config-if)# spanning-tree bpduglobal flooding
```

spanning-tree guard root

use the `spanning-tree guard root` Interface (Ethernet, Port Channel) Configuration mode command to enable Root Guard on all spanning-tree instances on the interface. Root guard prevents the interface from becoming the root port of the device. Use the **no** form of this command to disable the root guard on the interface.

Syntax

`spanning-tree guard root`

`no spanning-tree guard root`

Default Configuration

Root guard is disabled.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

Root Guard can be enabled when the device operates in any mode (STP, RSTP and MSTP).

When Root Guard is enabled, the port changes to the alternate state if the spanning-tree calculations select the port as the root port.

Example

The following example prevents te0/1 from being the root port of the device.

```
console(config)# interface te0/1
console(config-if)# spanning-tree guard root
```

spanning-tree bpduguard

Use the **spanning-tree bpduguard** Interface (Ethernet, Port Channel) Configuration 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

bpduguard *enable*—Enables BPDU Guard.

bpduguard *disable*—Disables BPDU Guard.

Default Configuration

BPDU Guard is disabled.

Command Mode

Interface (Ethernet, Port Channel) Configuration 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 te0/4 when it receives a BPDU.

```
console(config)# interface te0/4
console(config-if)# spanning-tree bpduguard enable
```

clear spanning-tree detected-protocols

Use the clear spanning-tree detected-protocols Privileged EXEC mode command to restart the STP migration process (force 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.

Default Configuration

All interfaces.

Command Mode

Privileged EXEC mode

User Guidelines

This feature can only be used when working in RSTP or MSTP mode.

Example

This restarts the STP migration process on all interfaces.

```
console# clear spanning-tree detected-protocols
```

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: 1–16)
- **priority**—Specifies the device priority for the specified spanning-tree instance. This setting determines 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 priority 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
```

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 BDPUs are 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

`max-hops hop-count`—Specifies the number of hops in an MST region before the BDPUs are 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
```

spanning-tree mst port-priority

Use the `spanning-tree mst port-priority` Interface (Ethernet, Port Channel) Configuration 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 is 128.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The priority value must be a multiple of 16.

Example

The following example configures the port priority of te0/1 to 144.

```
console(config)# interface te0/1
console(config-if)# spanning-tree mst 1 port-priority 144
```

spanning-tree mst cost

Use the **spanning-tree mst cost** Interface (Ethernet, Port Channel) Configuration mode command to configure the path cost for 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
```

Default Configuration

N/A

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
TenGigabit Ethernet (10000 Mbps)	2000	2
Gigabit Ethernet (1000 Mbps)	20,000	4
Ethernet (10 Mbps)	2,000,000	100

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

Example

The following example configures the MSTP instance 1 path cost for port te0/1 to 4.

```
console(config)# interface te0/1
console(config-if)# spanning-tree mst 1 cost 4
```

spanning-tree mst configuration

Use the **spanning-tree mst configuration** Global Configuration mode command to enable configuring an MST region by entering the 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 must 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
```

instance (MST)

Use **instance** MST Configuration mode command to map VLANs to an MST instance. Use the **no** form of this command to restore the 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
```

name (MST)

Use the **name** MST Configuration mode command to define the MST instance name. Use the **no** form of this command to restore the default setting.

Syntax

name *string*

no name

Parameters

string—Specifies the MST instance 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 instance name as Region1.

```
console(config)# spanning-tree mst configuration  
console(config-mst)# name region1
```

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
```

show (MST)

Use the **show** MST Configuration mode command to display 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.

Default Configuration

N/A

Command Mode

MST Configuration mode

Example

The following example displays a pending MST region configuration

```
console(config-mst)# show pending
Gathering information .....
Current MST configuration
Name: Region1
Revision: 1
Instance  VLANs Mapped          State
-----  -
0         1-4094                      Disabled
console(config-mst)#
```

exit (MST)

Use the **exit** MST Configuration mode command to exit the MST region Configuration mode and apply all configuration changes.

Syntax

exit

Parameters

N/A

Default Configuration

N/A

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  
console(config)#
```

abort (MST)

Use the **abort** MST Configuration mode command to exit the MST Configuration mode without applying the configuration changes.

Syntax

abort

Parameters

N/A

Default Configuration

N/A

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
```

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.

Default Configuration

If no interface is specified, the default is all interfaces.

Command Mode

Privileged EXEC mode

User Guidelines

This command only works when MST is enabled.

Example

The following examples display spanning-tree information in various configurations

```
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             te0/1

         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
te0/1	Enable	128.1		20000	FRW	Root	-----t	P2p (RSTP)
te0/2	d	128.2		20000	FRW	Desg	No	Shared
te0/3	Enable	128.3		20000	-	-	No	(STP)
te0/4	d	128.4		20000	BLK	Altn	-	-
	Disabl						No	Shared
	ed							(STP)
	Enable							
	d							

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 sec

Interfaces

Name	State	Prio.Nbr	Cost	Sts	Role	PortFas	Type
te0/1	--	128.1	20000	FRW	Desg	-----t	P2p (RSTP)
te0/2	Enable	128.2	20000	FRW	Desg	--	Shared
te0/3	d	128.3	20000	-	-	No	(STP)
te0/4	Enable	128.4	20000	FRW	Desg	No	-
	d					-	Shared
	Disabl						(STP)
	ed						
	Enable						
	d						

```
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 Priority 36864
ID 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
-	--	128.1	20000	-	-	-	-
te0/1	Enable	128.2	20000	-	-	--	-
te0/2	d	128.3	20000	-	-	-	-
te0/3	Enable	128.4	20000	-	-	-	-
te0/4	d					-	
	Disabl						
	ed						
	Enable						
	d						

```
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        te0/1

        Hello Time 2 sec          Max Age 20 sec          Forward Delay 15 sec
```

```
Bridge  Priority          36864
ID      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
-	--	128.1	20000	FRW	Root	-----t	P2p (RSTP)
te0/1	Enable	128.2	20000	FRW	Desg	--	Shared
te0/2	d	128.4	20000	BLK	Altn	No	(STP)
te0/4	Enable					No	Shared
	d					No	(STP)
	Enable						
	d						

```
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           te0/1

      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.Nbr	Cost	Sts	Role	PortFas	Type
-	-	128.4	19	BLK	Altn	-	Shared
te0/4	Enable					-	(STP)
	d					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        te0/1

      Hello Time 2 sec          Max Age 20      Forward Delay 15
                                sec                      sec
```

```
Bridge Priority          36864
ID 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 (te0/1) enabled
```

```
State: Forwarding
```

```
Role: Root
```

```
Port id: 128.1
```

```
Port cost: 20000
```

```
Type: P2p (configured: auto)
RSTP
```

```
Port Fast: No (configured:no)
```

```
Address: 00:01:42:97:e0:00
```

```
Designated bridge Priority:
32768
```

```
Designated path cost: 0
```

```
BPDU guard: Disabled
```

```
Designated port id: 128.25
```

```
Guard root: Disabled
```

Number of transitions to forwarding state: 1

BPDU: sent 2, received 120638

Port 2 (te0/2) 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
	BPDU guard: Disabled
Designated port id: 128.2	
Guard root: Disabled	

Number of transitions to forwarding state: 1

BPDU: sent 2, received 170638

Port 3 (te0/3) 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 (te0/4) enabled

State: Blocking	Role: Alternate
Port id: 128.4	Port cost: 20000
Type: Shared (configured:auto)	Port Fast: No (configured:no)
STP	Address: 00:30:94:41:62:c8
Designated bridge Priority: 28672	Designated path cost: 20000
	BPDU guard: Disabled
Designated port id: 128.25	
Guard root: Disabled	

Number of transitions to forwarding state: 1

BPDU: sent 2, received 120638

Number of transitions to forwarding state: N/A
BPDU: sent N/A, received N/A

console# **show spanning-tree ethernet** te0/1

Port 1 (te0/1) enabled
State: Forwarding Role: Root
Port id: 128.1 Port cost: 20000
Type: P2p (configured: auto) Port Fast: No (configured:no)
RSTP Address: 00:01:42:97:e0:00
Designated bridge Priority: Designated path cost: 0
32768 BPDU guard: Disabled
Designated port id: 128.25
Guard root: 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 te0/1

 Hello Time 2 secMax 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 secMax Age 20   Forward Delay 15
                        sec                   sec

                Max hops 20

```

Interfaces

Name	State	Prio.	Nbr	Cost	Sts	Role	PortFas	Type
te0/1	Enable	128.1		20000	FRW	Root	t	-
te0/2	d	128.2		20000	FRW	Desg	No	P2p Bound
te0/3	Enable	128.3		20000	FRW	Desg	No	(RSTP)
te0/4	d	128.4		20000	FRW	Desg	No	Shared
	Enable						No	Bound (STP)
	d							P2p
	Enable							P2p
	d							

MST 1 Vlans Mapped: 10-20

```

Root ID           Priority 24576
                  Address  00:02:4b:29:89:76
                  Path Cost 20000
                  Root Port  te0/4
                  Rem hops  19

```

```

Bridge ID         Priority 32768
                  Address  00:02:4b:29:7a:00

```


Interfaces

Name	State	Prio.Nbr	Cost	Sts	Role	PortFas	Type
----	-----	-----	-----	---	----	t	-----
te0/1	Enable	128.1	20000	FRW	Boun	-----	P2p Bound
te0/2	d	128.2	20000	FRW	Boun	No	(RSTP)
te0/3	Enable	128.3	20000	BLK	Altn	No	Shared
te0/4	d	128.4	20000	FRW	Root	No	Bound (STP)
	Enable					No	P2p
	d						P2p
	Enable						
	d						

```
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 te0/1
```

```
Hello Time 2 secMax Age 20 Forward Delay 15  
sec sec sec
```

```
IST Master ID    Priority 32768
```

```
Address 00:02:4b:29:7a:00
```

```
This switch is the IST master.
```

```
Hello Time 2 secMax Age 20 Forward Delay 15  
sec 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 (te0/1) enabled
State: Forwarding Role: Root
Port id: 128.1 Port cost: 20000
Type: P2p (configured: auto) Boundary RSTP Port Fast: No
(configured:no)
Designated bridge Priority: 32768 Address:
Designated port id: 128.25 00:01:42:97:e0:00
Number of transitions to forwarding state: 1 Designated path cost: 0
BPDU: sent 2, received 120638

Port 2 (te0/2) enabled
State: Forwarding Role: Designated
Port id: 128.2 Port cost: 20000
Type: Shared (configured: auto) Boundary STP Port Fast: No
(configured:no)
Designated bridge Priority: 32768 Address:
Designated port id: 128.2 00:02:4b:29:7a:00
Number of transitions to forwarding state: 1 Designated path cost:
20000
BPDU: sent 2, received 170638

Port 3 (te0/3) enabled
State: Forwarding Role: Designated
Port id: 128.3 Port cost: 20000
Type: Shared (configured: auto) Internal Port Fast: No
(configured:no)
Designated bridge Priority: 32768 Address:
Designated port id: 128.3 00:02:4b:29:7a:00
Number of transitions to forwarding state: 1 Designated path cost:
20000
BPDU: sent 2, received 170638

```
Port 4 (te0/4) enabled
State: Forwarding                               Role: Designated
Port id: 128.4                                  Port cost: 20000
Type: Shared (configured: auto)                 Port Fast: No
Internal                                         (configured:no)
Designated bridge Priority: 32768               Address:
Designated port id: 128.2                       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 te0/4
                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
```

```
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  te0/1

                 Hello Time 2 secMax Age 20   Forward Delay 15
                               sec           sec

                 Max hops 20
```

show spanning-tree bpdud

Use the `show spanning-tree bpdud` User EXEC mode command to display the BPDUD handling when spanning-tree is disabled.

Syntax

```
show spanning-tree bpdud [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.

Default Configuration

Show information for all interfaces. If `detailed` is not used, only present ports are displayed.

Command Mode

User EXEC mode

Example

The following examples display spanning-tree BPDU information:

```
console# show spanning-tree bpdu
```

The following is the output if the global BPDU handling command is not supported.

Interface	Admin Mode	Oper Mode
te0/1	Filtering	Filtering
te0/2	Filtering	Filtering
te0/3	Filtering	Guard

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
te0/1	Global	Flooding
te0/2	Global	STP
te0/3	Flooding	STP

VLAN Commands

vlan database

Use the `vlan database` Global Configuration mode command to enter the VLAN Configuration mode. This mode is used to create VLAN(s) and define 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
```

vlan

Use the **vlan** VLAN Configuration mode or Global Configuration mode command to create a VLAN and assign it a name (if only a single VLAN is being created). Use the **no** form of this command to delete the VLAN(s).

Syntax

```
vlan vlan-range | { vlan-id [name vlan-name] }
```

```
no vlan vlan-range
```

Parameters

- *vlan-range*—Specifies a list of VLAN IDs. Separate nonconsecutive VLAN IDs with a comma and no spaces. Use a hyphen to designate a range of IDs (range: 2-4094).
- *vlan-id*—Specifies a VLAN ID. (range: 2-4094).
- *vlan-name*—Specifies the VLAN name. (range: 1–32 characters).

Default Configuration

VLAN 1 exists by default.

Command Mode

Global Configuration mode

VLAN Database Configuration mode

User Guidelines

If the VLAN does not exist, it is created. If the VLAN cannot be created then the command is finished with error and the current context is not changed.

Example

The following example creates a few VLANs. VLAN 1972 is assigned the name Marketing.

```
console(config)# vlan database
console(config-vlan)# vlan 19-23
console(config-vlan)# vlan 100
```

```
console(config-vlan)# vlan 1972 name Marketing
console(config-vlan)# exit
```

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 is dynamic or permanent
- 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
```

Created by: D-Default, S-Static, G-GVRP, R-Radius Assigned VLAN

VLAN	Name	Ports	Created by
1	Default	te0/1	D
10	Marketing	te0/2	S
91	11	te0/2	SGR
92	11	te0/3-4	G
93	11	te0/3-4	GR

Example 2—The following example displays information for the default VLAN (VLAN 1):

```
console# show vlan tag 1
```

Created by: D-Default, S-Static, G-GVRP, R-Radius Assigned VLAN

VLAN	Name	Ports	Created by
1	Default	te0/1-2	D

Example 3—The following example displays information for the VLAN named Marketing:

```
console# show vlan name Marketing
```

Created by: D-Default, S-Static, G-GVRP, R-Radius Assigned VLAN

VLAN	Name	Ports	Created by
10	Marketing	te0/3-4	S

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.

Syntax

```
interface vlan vlan-id
```

Parameters

- *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, this command is finished with an error and the current context is not changed.

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
```

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-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.

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
```

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.

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 (VLAN) Configuration mode

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
```

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 (Ethernet, Port Channel) Configuration mode

User Guidelines

Note that packets are subject to all filtering rules and Filtering Database (FDB) decisions.

Use this command to isolate Unicast, Multicast, and Broadcast traffic at Layer 2 from other protected ports (that are not associated with the same community as the ingress interface) on the same switch. Please note that the packet is still subject to FDB decision and to all filtering rules. Use the `switchport community` Interface Configuration command to associate the interface with a community.

Example

```
console(config)# interface te0/1
console(config-if)# switchport protected-port
```

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 | 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.

Default Configuration

Show all protected interfaces. If `detailed` is not used, only present ports are displayed.

Command Mode

User EXEC mode

Example

```
console# show interfaces protected-ports
```

Interface	State	Community
-----	-----	-----
te0/1	Protected	1
te0/2	Protected	Isolated
te0/3	Unprotected	20
te0/4	Unprotected	Isolated

switchport community

Use the **switchport community** Interface Configuration mode command to associate a protected port with a community. Use the **no** form of this command to return to the default.

Syntax

```
switchport community community
```

```
no switchport community
```

Parameters

- *community*—Specifies the community number. (range: 1 - 31).

Default Configuration

The port is not associated with a community.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The command is relevant only when the port is defined as a protected port. Use the **switchport protected-port** Interface Configuration command to define a port as a protected port.

Example

```
console(config)# interface te0/1
console(config-if)# switchport community 1
```

switchport

Use the **switchport** Interface Configuration mode command to put an interface that is in Layer 3 mode into Layer 2 mode. Use the **no** form of this command to put an interface in Layer 3 mode.

Syntax

switchport

no switchport

Parameters

N/A

Default Configuration

Layer 2 mode

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

Use the **no switchport** command to set the interface as a Layer 3 interface

Examples:

Example 1 - The following example puts the port **te0/1** into Layer 2 mode.

```
console(config)# interface te0/1
console(config-if)# switchport
```

Example 2 - The following example puts the port **te0/1** into Layer 3 mode.

```
console(config)# interface te0/1
```

```
console(config-if)# no switchport
```

switchport mode

Use the **switchport mode** Interface Configuration mode command to configure the VLAN membership mode. Use the **no** form of this command to restore the default configuration.

Syntax

```
switchport mode {access | trunk | general | private-vlan {promiscuous | host} | 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.
- **private-vlan promiscuous**—Private-VLAN promiscuous port.
- **private-vlan host**—Private-VLAN host port.

Default Configuration

Access mode.

Command Mode

Interface (Ethernet, Port Channel) Configuration 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.

Example

Example 1 - The following example configures te0/1 as an access port (untagged layer 2) VLAN port.

```
console(config)# interface te0/1
console(config-if)# switchport mode access
console(config-if)# switchport access vlan 2
```

Example 2 - The following example puts the port te0/2 into private-vlan host mode.

```
console(config)# interface te0/2
console(config-if)# switchport mode private-vlan host
```

switchport access vlan

A port in access mode can be an untagged member of at most a single VLAN. The **switchport access vlan** Interface Configuration command reassigns an interface to a different VLAN than it currently belongs or assigns it to **none**, in which case it is not a member of any VLAN.

The **no** form of this command to restore the default configuration.

Syntax

```
switchport access vlan {vlan-id | none}
```

```
no switchport access vlan
```

Parameters

- *vlan-id*—Specifies the VLAN to which the port is configured.
- **none**—Specifies that the access port cannot belong to any VLAN.

Default Configuration

The interface belongs to the Default VLAN.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

When the port is assigned to a different VLAN, it is automatically removed from its previous VLAN and added to the new VLAN. If the port is assigned to **none**, it is removed from the previous VLAN and not assigned to any other VLAN.

A non-existent VLAN can be assigned as an Access VLAN. If the Access VLAN does not exist the **show interfaces switchport** command adds text "(Inactive)" after VLAN ID.

Example

The following example assigns access port te0/1 to VLAN 2 (and removes it from its previous VLAN).

```
console(config)# interface te0/2
console(config-if)# switchport mode access
console(config-if)# switchport access vlan 2
```

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. Use the **switchport trunk allowed vlan** Interface Configuration mode command to add/remove VLAN(s) to/from a trunk port.

Syntax

```
switchport trunk allowed vlan {all | none | add vlan-list | remove vlan-list | except vlan-list}
```

Parameters

- **all**—Specifies all VLANs from 1 to 4094. At any time, the port belongs to all VLANs existing at the time. (range: 1–4094).
- **none**—Specifies an empty VLAN list The port does not belong to any VLAN.

- **add** *vlan-list*—List of VLAN IDs to add to the port. Separate nonconsecutive VLAN IDs with a comma and no spaces. Use a hyphen to designate a range of IDs.
- **remove** *vlan-list*—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.
- **except** *vlan-list*—List of VLAN IDs including all VLANs from range 1-4094 except VLANs belonging to *vlan-list*.

Default Configuration

By default, trunk ports belongs to all created VLANs.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

Use the **switchport trunk allowed vlan** command to specify which VLANs the port belongs to when its mode is configured as trunk.

Example

To add VLANs 2,3 and 100 to trunk ports 1 to 13

```
console(config)# interface range te0/1-3
console(config-if)# switchport mode trunk
console(config-if)# switchport trunk allowed vlan add 2-3,100
console(config-if)# exit
```

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 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 | none }
```

no switchport trunk native vlan

Parameters

- *vlan-id*—Specifies the native VLAN ID.
- **none**—Specifies the access port cannot belong to any VLAN.

Default Configuration

The default native VLAN is the Default VLAN.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

If the interface does not belong to the VLAN it is added to the VLAN, if the interface is not a forbidden member of this VLAN. If the interface is a forbidden member of an added VLAN, the interface does not become a member of this specific VLAN and an error message is displayed. This message will only be displayed once.") and the command continues to execute if there are more VLANs in the *vlan-list*.

The interface is set as VLAN untagged egress interface. A value of the interface PVID is set to this VLAN ID.

The configuration is applied only when the port mode is trunk.

Examples

The following example defines VLAN 2 as native VLAN for port *te0/1*:

```
console(config)# interface te0/1
console(config-if)# switchport trunk native vlan 2
console(config-if)# exit
```

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 | remove} vlan-list [tagged | untagged]
```

```
no switchport general allowed vlan
```

Parameters

- **add** *vlan-list*—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: 1–4094)
- **remove** *vlan-list*—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.
- **tagged**—Specify that packets are transmitted tagged for the configured VLANs
- **untagged**—Specify that packets are transmitted untagged for the configured VLANs (this is the default)

Default Configuration

The port is not a member of any VLAN.

Packets are transmitted untagged.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

If the interface is a forbidden member of an added VLAN, the interface does not become a member of this specific VLAN. There will be an error message in this case ("An interface cannot become a member of a forbidden VLAN. This message will only be displayed once.") and the command continues to execute in case if there are more VLANs in the *vlan-list*.

The configuration is applied only when the port mode is general.

Example

The example adds te0/1 and to VLAN 2 and 3. Packets are tagged on the egress:

```
console(config)# interface te0/1
console(config-if)# switchport general allowed vlan add 2-3
tagged
```

switchport general pvid

Use the `switchport general pvid` Interface Configuration 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

- *vlan-id*—Specifies the Port VLAN ID (PVID).

Default Configuration

The PVID is the Default VLAN PVID.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

Example

Example 1 - The following example sets the te0/2 PVID to 234.

```
console(config)# interface te0/2
console(config-if)# switchport general pvid 234
```

Example 2 - The following example performs the following:

- Adds VLANs 2&3 as tagged, and VLAN 100 as untagged to te0/4

- Defines VID 100 as the PVID
-

```
console(config)# interface te0/4
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)# exit
```

switchport general ingress-filtering disable

Use the `switchport general ingress-filtering disable` Interface Configuration 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 (Ethernet, Port Channel) Configuration mode

Example

The following example disables port ingress filtering on te0/1.

```
console(config)# interface te0/1
console(config-if)# switchport mode general
```

```
console(config-if)# switchport general ingress-filtering  
disable
```

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 (Ethernet, Port Channel) Configuration mode

Example

The following example configures port te0/3 to be in general mode and to discard untagged frames at ingress.

```
console(config)# interface te0/3  
console(config-if)# switchport mode general  
console(config-if)# switchport general acceptable-frame-type  
tagged-only
```

switchport customer vlan

Use the `switchport customer vlan` Interface Configuration mode command to set the port's VLAN when the interface is in customer mode (set by the `switchport mode` command). Use the `no` form of this command to restore the default configuration.

Syntax

```
switchport customer vlan vlan-id
```

```
no switchport customer vlan
```

Parameters

- *vlan-id*—Specifies the customer VLAN.

Default Configuration

No VLAN is configured as customer.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

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.

Example

The following example defines te0/4 as a member of customer VLAN 5.

```
console(config)# interface te0/4
console(config-if)# switchport mode customer
console(config-if)# switchport customer vlan 5
```

switchport general forbidden vlan

Use the `switchport general forbidden vlan` Interface Configuration mode command to forbid adding/removing specific VLANs to/from a port. Use the `no` form of this command to restore the default configuration.

Syntax

```
switchport general forbidden vlan {add vlan-list | remove vlan-list}
```

```
no switchport general forbidden vlan
```

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.
- `remove vlan-list`—Specifies a list of VLAN IDs to remove from interface. 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 (Ethernet, Port Channel) Configuration mode

User Guidelines

The forbidden VLAN cannot be one that does not exist on the system, or one that is already defined on the port.

Example

The following example define s te0/4 as a forbidden membership in VLANs 5-7:

```
console(config)# interface te0/4
console(config-if)# switchport general forbidden vlan add 5-7
console(config-if)# exit
```

map protocol protocols-group

Use the `map protocol protocols-group` VLAN Configuration mode command to map a protocol to a group of protocols. This protocol group can then be used in `switchport general map protocols-group vlan`. Use the `no` form of this command to delete a protocol from a group.

Syntax

`map protocol protocol [encapsulation-value] protocols-group group`

`no map protocol protocol [encapsulation]`

Parameters

- *protocol*—Specifies a 16-bit protocol number or one of the reserved names listed in the User Guidelines. (range: 0x0600–0xFFFF)
- *encapsulation-value*—Specifies one of the following values: Ethernet, rfc1042, llcOther.
- *protocols-group group*—Specifies the group number of the group of protocols (range: 1–2147483647).

Default Configuration

The default encapsulation value is Ethernet.

Command Mode

VLAN Database Configuration mode

User Guidelines

Forwarding of packets based on their protocol requires setting up groups of protocols and then mapping these groups to VLANs.

The value 0x8100 is not valid as the protocol number for Ethernet encapsulation.

The following protocol names are reserved for Ethernet Encapsulation:

- ip
- arp
- ipv6

- ipx

Example

The following example maps the IP protocol to protocol group number 213.

```
console(config)# vlan database
console(config-vlan)# map protocol ip protocols-group 213
```

switchport general map protocols-group vlan

Use the `switchport general map protocols-group vlan` Interface Configuration mode command to forward packets based on their protocol, otherwise known as setting up a classifying rule. This command forwards packets arriving on an interface containing a specific protocol to a specific VLAN. Use the `no` form of this command to stop forwarding packets based on their protocol.

Syntax

```
switchport general map protocols-group group vlan vlan-id
```

```
no switchport general map protocols-group group
```

Parameters

- *group*—Specifies the group number as defined in `map protocol protocols-group` command (range: 1–65535).
- *vlan-id*—Defines the VLAN ID in the classifying rule.

Default Configuration

N/A

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The VLAN classification rule priorities are:

- MAC-based VLAN (best match among the rules)

- Subnet-based VLAN (best match among the rules)
- Protocol-based VLAN
- PVID

Example

The following example forwards packets with protocols belong to protocol-group 1 to VLAN 8.

```
console(config-if)# switchport general map protocols-group 1  
vlan 8
```

show vlan protocols-groups

Use the `show vlan protocols-groups EXEC` mode command to display the protocols that belong to the defined protocols-groups.

Syntax

`show vlan protocols-groups`

Parameters

N/A

Default Configuration

N/A

Command Mode

User EXEC mode

Example

The following example displays protocols-groups information.

```
console# show vlan protocols-groups
```

Encapsulation	Protocol	Group ID
-----	-----	-----
Ethernet	0x800 (IP)	1
Ethernet	0x806 (ARP)	1
Ethernet	0x86dd (IPv6)	2
Ethernet	0x8898	3

map subnet subnets-group

Use the `map subnet subnets-group` VLAN Configuration mode command to map an IP subnet to a group of IP subnets. Use the `no` form of this command to delete the map.

Syntax

```
map subnet ip-address prefix-mask subnets-group group
```

```
no map subnet ip-address prefix-mask
```

Parameters

- *ip-address*—Specifies the IP address prefix of the subnet to be mapped to the group.
- *prefix-mask*—Specifies the number of 1s in the mask.
- *group*—Specifies the group number. (range: 1–2147483647)

Default Configuration

N/A

Command Mode

VLAN Database Configuration mode

User Guidelines

Forwarding of packets based on their IP subnet requires setting up groups of IP subnets and then mapping these groups to VLANs.

Example

The following example maps an IP subnet to the group of IP subnets 4. It then maps this group of IP subnets to VLAN 8

```
console(config)# vlan database
console(config-vlan)# map subnet 172.16.1.1 24 subnets-group
4
console(config-vlan)# switchport general map subnets-group 4
vlan 8
```

switchport general map subnets-group vlan

Use the `switchport general map subnets-group vlan` Interface Configuration mode command to set a subnet-based classification rule. Use the `no` form of this command to delete a subnet-based classification rule.

Syntax

```
switchport general map subnets-group group vlan vlan-id
no switchport general map subnets-group group
```

Parameters

- *group*—Specifies the group number. (range: 1–2147483647)
- *vlan-id*—Defines the VLAN ID associated with the rule.

Default Configuration

N/A

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

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 maps an IP subnet to the group of IP subnets 4. It then maps this group of IP subnets to VLAN 8

```
console(config)# vlan database
console(config-vlan)# map subnet 172.16.1.1 24 subnets-group
4
console(config-vlan)# switchport general map subnets-group 4
vlan 8
```

show vlan subnets-groups

Use the `show vlan subnets-groups` EXEC mode command to display subnets-groups information.

Syntax

```
show vlan subnets-groups
```

Parameters

N/A

Default Configuration

N/A

Command Mode

User EXEC mode

Example

The following example displays subnets-groups information.

```
console# show vlan subnets-groups
IP Subnet Address      Mask      Group ID
-----
      1.1.1.1          32        1
     172.16.2.0       24        2
```

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*—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or port-channel.

Command Mode

Privileged EXEC mode

Default

Displays the status of all interfaces.

Example

```
console# show interfaces switchport te0/1
Gathering information...
Name: te0/1
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
General GVRP VLANs: none
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
Protected: Enabled, Uplink is te0/1
Classification rules:
Classification Type   Group ID   VLAN ID
-----
Protocol              1         19
Protocol              1         20
Protocol              2         72
Subnet                1         15

```

private-vlan

Use the **private-vlan** Interface VLAN Configuration mode command to configure a private VLAN. Use the **no** form of this command to return the VLAN to normal VLAN configuration.

Syntax

```
private-vlan {primary | community | isolated}
```

no private-vlan

Parameters

- **primary**—Designate the VLAN as a primary VLAN.
- **community**—Designate the VLAN as a community VLAN.
- **isolated**—Designate the VLAN as an isolated VLAN.

Default Configuration

No private VLANs are configured.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

- The VLAN type cannot be changed if there is a private VLAN port that is a member in the VLAN.
- The VLAN type cannot be changed if it is associated with other private VLANs.
- The VLAN type is not kept as a property of the VLAN when the VLAN is deleted.

Example

The following example set vlan 2 to be primary vlan:

```
console(config)# interface vlan 2  
console(config-if)# private-vlan primary
```

private-vlan association

Use the **private-vlan association** Interface VLAN Configuration mode command to configure the association between the primary VLAN and secondary VLANs. Use the **no** form of this command to remove the association.

Syntax

`private-vlan association [add | remove] secondary-vlan-list`

`no private-vlan association`

Parameters

- **add** *secondary-vlan-list*—List of VLAN IDs of type secondary to add to a primary VLAN. Separate nonconsecutive VLAN IDs with a comma and no spaces. Use a hyphen to designate a range of IDs. This is the default action.
- **remove** *secondary-vlan-list*—List of VLAN IDs of type secondary to remove association from a primary VLAN. Separate nonconsecutive VLAN IDs with a comma and no spaces. Use a hyphen to designate a range of IDs.

Default Configuration

No private VLANs are configured.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

- The command can only be executed in the context of the primary VLAN.
- A private VLAN cannot be removed or have its type changed, if it is associated with other private VLANs.
- A primary VLAN can be associated with only a single, isolated VLAN.
- A secondary VLAN can be associated with only one primary VLAN.
- The association of secondary VLANs with a primary VLAN cannot be removed if there are private VLAN ports that are members in the secondary VLAN.
- In MSTP mode, all the VLANs that are associated with a private VLAN must be mapped to the same instance.

Example

The following example associate secondary VLAN 20,21,22 and 24 to primary VLAN 2.

```
console(config)# interface vlan 2  
console(config-if)# private-vlan association add 20-22,24
```

switchport private-vlan mapping

Use the `switchport private-vlan mapping` Interface Configuration mode command to configure the VLANs of the private VLAN promiscuous port. Use the `no` form of this command to reset to default.

Syntax

`switchport private-vlan mapping primary-vlan-id [add | remove] secondary-vlan-list`

`no switchport private-vlan mapping`

Parameters

- *primary-vlan-id*—The VLAN ID of the primary VLAN.
- `add secondary-vlan-list`—Specifies one or more secondary VLANs to be added to the port.
- `remove secondary-vlan-list`—Specifies one or more secondary VLANs to be removed from the port.

Default Configuration

No VLAN is configured.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The secondary VLANs should be associated with the primary VLANs, otherwise the configuration is not accepted.

Example

The following example add promiscuous port te0/1 to primary VLAN 10 and to secondary VLAN 20.

```
console(config)# interface te0/4
console(config-if)# switchport private-vlan mapping 10 add 20
```

switchport private-vlan host-association

Use the `switchport private-vlan host-association` Interface Configuration mode command to configure the VLANs of the private VLAN host port. Use the `no` form of this command to reset to default.

Syntax

```
switchport private-vlan host-association primary-vlan-id secondary-vlan-id
no switchport private-vlan host-association
```

Parameters

- *primary-vlan-id*—The VLAN ID of the primary VLAN.
- *secondary-vlan-list*—Specifies the secondary VLAN. The secondary VLAN is an isolated port.

Default Configuration

No VLAN is configured.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The secondary VLAN must be associated with the primary VLAN, otherwise the configuration is not accepted. See the [private-vlan association](#) command.

Example

The following example set port te0/4 to secondary VLAN 20 in primary VLAN 10.

```
console(config)# interface te0/1
console(config-if)# switchport private-vlan host-association
10 20
```

show vlan private-vlan

Use the `show vlan private-vlan` EXEC mode command to display private VLAN information.

Syntax

```
show vlan private-vlan [tag vlan-id]
```

Parameters

- `tag vlan-id`—Primary VLAN that represent the private VLAN to be displayed.

Default Configuration

All private VLANs are displayed.

Command Mode

User EXEC mode

User Guidelines

The `show` command does not include non-private VLAN ports that are members in private VLANs. Tag parameters of non-primary VLAN will result in an empty show output.

Example

```
console# show vlan private-vlan
Primary      Secondary    Type          Ports
-----
```

150		primary	te0/1
150	151	isolated	te0/2
160		primary	te0/3
160	161	community	te0/4

```
console# show vlan private-vlan 150
```

Primary	Secondary	Type	Ports
-----	-----	-----	-----
150		primary	te0/1
150	151	isolated	te0/4

switchport access multicast-tv vlan

Use the `switchport access multicast-tv vlan` Interface Configuration mode command to enable receiving Multicast transmissions on an interface that is not the access port VLAN, while keeping the L2 segregation with subscribers on different access port VLANs. Use the `no` form of this command to disable receiving Multicast transmissions.

Syntax

```
switchport access multicast-tv vlan vlan-id
```

```
no switchport access multicast-tv vlan
```

Parameters

- *vlan-id*—Specifies the Multicast TV VLAN ID.

Default Configuration

Receiving Multicast transmissions is disabled.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The user cannot transmit Multicast transmissions on the Multicast TV VLAN.

A Multicast TV VLAN cannot be enabled if a guest VLAN is enabled on the interface.

Example

The following example enables te0/4 to receive Multicast transmissions from VLAN 11.

```
console(config)# interface te0/4
console(config-if)# switchport access multicast-tv vlan 11
```

switchport customer multicast-tv vlan

Use the `switchport customer multicast-tv vlan` Interface Configuration mode command to enable receiving Multicast transmissions from a VLAN that is not the customer port's VLAN, while keeping the L2 segregation with subscribers on different customer port VLANs.

Syntax

```
switchport customer multicast-tv vlan {add vlan-list | remove vlan-list}
```

Parameters

- `add vlan-list`—Specifies a list of Multicast TV VLANs to add to interface.
- `remove vlan-list`—Specifies a list of Multicast TV VLANs to remove from interface.

Default Configuration

The port is not a member in any Multicast TV VLAN.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The user cannot transmit Multicast transmissions on Multicast TV VLANs.

A Multicast TV VLAN cannot be enabled if a guest VLAN is enabled on an interface.

Example

The following example enables te0/4 to receive Multicast transmissions from VLANs 5, 6, 7.

```
console(config)# interface te0/4
console(config-if)# switchport customer multicast-tv vlan
add 5-7
```

show vlan multicast-tv

Use the **show vlan Multicast-tv EXEC** mode command to display the source and receiver ports of Multicast-TV VLAN. Source ports can transmit and receive traffic to/from the VLAN, while receiver ports can only receive traffic from the VLAN.

Syntax

```
show vlan Multicast-tv vlan vlan-id
```

Parameters

- *vlan-id*—Specifies the VLAN ID.

Default Configuration

N/A

Command Mode

User EXEC mode

Example

The following example displays information on the source and receiver ports of Multicast-TV VLAN 1000.

```
console# show vlan multicast-tv vlan 1000

Source Ports          Receiver Ports
-----
te0/3, te0/4         te0/1-2
```

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 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 (Ethernet, Port Channel) Configuration mode

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 te0/3.

```
console(config)# interface te0/3
console(config-if)# ip internal-usage-vlan 200
```

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 the VLANs used internally by the device.

```
console# show vlan internal usage
```

Usage	VLAN	Reserved	IP address
-----	-----	-----	-----
te0/1	1007	No	Active
te0/2	1008	Yes	Inactive
te0/3	1009	Yes	Active

Voice VLAN Commands

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 | disabled }`

`no voice vlan state`

Parameters

- `oui-enabled`—Voice VLAN is of type OUI.
- `disabled`—Voice VLAN is disabled.

Default Configuration

Disabled

Command Mode

Global Configuration mode

User Guidelines

If the administrative state is:

- `disabled` — The operational state is `disabled`.
- `oui-enabled` —The operational state is `oui-enabled`.

Example:

Example 1 —The following example enables the OUI mode of Voice VLAN. The first try did not work - it was necessary to first disable voice VLAN.

```
console(config)# voice vlan state oui-enabled
```

Disable the voice VLAN before changing the voice VLAN trigger.

```
console(config)# voice vlan state disabled  
console(config)# voice vlan state oui-enabled  
<CR>
```

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 35 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 35
was created.
console(config)# 30-Apr-2011 00:19:51 %VLAN-I-ReceivedFromVSDP:
Voice VLAN updated by VSDP. Voice VLAN-ID 35, VPT 5, DSCP 46

```

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
00:01:81	Nortel
00:01:e3	Siemens AG Phone
00:03:6b	Cisco Phone
00:09:6e	Avaya Phone
00:0f:e2	Huawei-3COM Phone
00:10:49	Shoretel

OUI	Description
00:60:b9	NEC/Philips Phone
00:d0:1e	Pingtel Phone
00:e0:75	Veritel Polycom Phone
00:e0:bb	3COM 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 experimental
```

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

Interface Configuration mode

Example

The following example applies QoS attributes to voice packets.

```
console(config-if)# voice vlan cos mode all
```

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
```

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
```

voice vlan enable

Use the **voice vlan enable** Interface Configuration mode 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 mode

User Guidelines

This command is applicable only if the voice VLAN state is globally configured as OUI voice VLAN (using [voice vlan state](#)).

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 `te0/2`.

```
console(config)# interface te0/2
console(config-if)# voice vlan enable
```

voice vlan secure

Use the `voice vlan secure` Interface Configuration mode mode command to enable the secure mode for the OUI voice VLAN. Use the `no` form of this command to disable the secure mode (see User Guidelines for an explanation of secure mode).

Syntax

`voice vlan secure`

`no voice vlan secure`

Default Configuration

Disabled

Command Mode

Interface Configuration mode

User Guidelines

Secure mode specifies that packets that are classified to the voice VLAN with a source MAC address that is not a OUI address (defined by [voice vlan oui-table](#)) are discarded.

This command is relevant only to ports that were added to the voice VLAN automatically.

Example

The following example enables the secure mode for the OUI Voice VLAN on te0/4.

```
console(config)# interface te0/4
console(config-if)# voice vlan secure
```

show voice vlan

Use the `show voice vlan` Privileged 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* [{*interface-id* / *detailed*}]]

Parameters

- **type oui**—Common and OUI-voice-VLAN specific parameters are displayed.
- **interface-id**—Specifies an Ethernet port ID.
- **detailed**—Displays information for non-present ports in addition to present ports.

Default Configuration

If the **type** parameter is omitted the current Voice VLAN type is used.

If the **interface-id** parameter is omitted then information about all present interfaces is displayed. If **detailed** is used, non-present ports are also displayed.

Command Mode

Privileged EXEC mode

User Guidelines

Using this command without parameters displays the current voice VLAN type parameters and local and agreed voice VLAN settings.

Using this command with the **type** parameter displays the voice VLAN parameters relevant to the type selected. The the local and agreed voice VLAN settings are displayed only if this is the current voice VLAN state.

The **interface-id** parameter is relevant only for the OUI VLAN type.

Examples:

The following example displays the voice VLAN parameters.

```
switch>show voice vlan
Administrate Voice VLAN state is oui-enabled
The Operational Voice VLAN-ID is 2
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
te0/1	Yes	Yes	Yes	all
te0/2	Yes	Yes	No	src
te0/3	No	No		
...				

IGMP Snooping Commands

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
```

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-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 Snooping are supported.

To activate IGMP snooping, bridge multicast filtering must be enabled by the **bridge multicast filtering** command.

The user guidelines of the **bridge multicast mode** 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
```

ip igmp snooping vlan mrouter

Use the **ip igmp snooping vlan 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-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
```

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-id*—Specifies the VLAN.
- *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  
te0/1
```

ip igmp snooping vlan forbidden mrouter

Use the `ip igmp snooping vlan forbidden mrouter` 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-id*—Specifies the VLAN.
- *interface-list*—Specifies a list of interfaces. The interfaces can be of 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 te0/1
```

ip igmp snooping vlan static

Use the `ip igmp snooping vlan static` Global Configuration mode command to register an IP-layer Multicast address to the bridge table, and to add static ports to the group defined by this address. 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-id*—Specifies the VLAN.
- *ip-address*—Specifies the IP Multicast address.
- *interface-list*—Specifies a list of interfaces. The interfaces can be of 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 te0/1
```

ip igmp snooping vlan multicast-tv

Use the **ip igmp snooping vlan multicast-tv** Global Configuration mode command to define the Multicast IP addresses that are associated with a Multicast TV VLAN. Use the **no** form of this command to remove all associations.

Syntax

```
ip igmp snooping vlan vlan-id multicast-tv ip-multicast-address [count
number]
```

```
no ip igmp snooping vlan vlan-id multicast-tv ip-multicast-address [count
number]
```

Parameters

- *vlan-id*—Specifies the VLAN
- *ip-multicast-address*—Multicast IP address
- **count** *number*—Configures multiple contiguous Multicast IP addresses. If not specified, the default is 1. (Range: 1–256)

Default Configuration

No Multicast IP address is associated.

Command Mode

Global Configuration mode

User Guidelines

Use this command to define the Multicast transmissions on a Multicast-TV VLAN. The configuration is only relevant for an Access port that is a member in the configured VLAN as a Multicast-TV VLAN.

If an IGMP message is received on such an Access port, it is associated with the Multicast-TV VLAN only if it is for one of the Multicast IP addresses that are associated with the Multicast-TV VLAN.

Up to 256 VLANs can be configured.

Example

```
console(config)# ip igmp snooping vlan 1 multicast-tv
239.2.2.2 count 3
```

ip igmp snooping querier

Use the `ip igmp snooping querier` Global Configuration mode command to enable globally the IGMP Snooping querier. Use the `no` form of this command to disable the IGMP Snooping querier globally.

Syntax

`ip igmp snooping querier`

`no ip igmp snooping querier`

Parameters

N/A

Default Configuration

Enabled

Command Mode

Global Configuration mode

User Guidelines

To run the IGMP Snooping querier on a VLAN, you have enable it globally and on the VLAN.

Example

The following example disables the IGMP Snooping querier globally:

```
console(config)# no ip igmp snooping querier
```

ip igmp snooping querier address

Use the `ip igmp snooping querier address` Global Configuration mode command to define globally 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 querier address ip-address`

`no ip igmp snooping querier address`

Parameters

- *ip-address*—Source IP address.

Default Configuration

no IP address

Command Mode

Global Configuration mode

User Guidelines

If an IP address is configured for the VLAN, it is used as the source address of the IGMP snooping querier. If it is not configured for the VLAN and an IP address is configured globally, it is used as the source address of the IGMP snooping querier. If an IP address is not configured for the VLAN and is not configured globally, the minimum IP address defined on the VLAN is used.

If an IP address is not configured for the VLAN and is not configured globally and no IP address is defined on the VLAN, the querier is disabled.

Example

The following example define IP address 10.5.234.205 as the Querier Snooping IP address on a VLAN if it is not configured for the VLAN

```
console(config)# ip igmp snooping querier address
10.5.234.205
```

ip igmp snooping vlan querier

Use the `ip igmp snooping vlan querier` Global Configuration mode command to enable the IGMP Snooping querier on a specific VLAN. Use the `no` form of this command to disable the IGMP Snooping querier on a VLAN interface.

Syntax

```
ip igmp snooping vlan vlan-id querier
no ip igmp snooping vlan vlan-id querier
```

Parameters

- *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.

Example

The following example enables the IGMP Snooping querier on VLAN 1:

```
console(config)# ip igmp snooping vlan 1 querier
```

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-id*—Specifies the VLAN.
- *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
```

ip igmp snooping vlan querier election

Use the `ip igmp snooping vlan querier election` Global Configuration mode command to enable IGMP Querier election mechanism of an IGMP

Snooping querier on a specific VLAN. Use the **no** form of this command to disable Querier election mechanism.

Syntax

```
ip igmp snooping vlan vlan-id querier election
```

```
no ip igmp snooping vlan vlan-id querier election
```

Parameters

- *vlan-id*—Specifies the VLAN.

Default Configuration

Enabled

Command Mode

Global Configuration mode

User Guidelines

Use the **no** form of the **ip igmp snooping vlan querier election** command to disable IGMP Querier election mechanism on a VLAN.

If the IGMP Querier election mechanism is enabled, the IGMP Snooping querier supports the standard IGMP Querier election mechanism specified in RFC2236 and RFC3376.

If IGMP Querier election mechanism is disabled, IGMP Snooping Querier delays sending General Query messages for 60 seconds from the time it was enabled. During this time, if the switch did not receive an IGMP query from another Querier - it starts sending General Query messages. Once the switch acts as a Querier, it will stop sending General Query messages if it detects another Querier on the VLAN. In this case, the switch will resume sending General Query messages if it does hear another Querier for Query Passive interval that equals:

$$\langle \text{Robustness} \rangle * \langle \text{Query Interval} \rangle + 0.5 * \langle \text{Query Response Interval} \rangle.$$

See the [ip igmp robustness](#), [ip igmp last-member-query-interval](#), and [ip igmp query-max-response-time](#) commands for configurations of these parameters.

It is recommended to disable IGMP Querier election mechanism if there is an IPM Multicast router on the VLAN.

Example

The following example disables IGMP Snooping Querier election on VLAN 1:

```
console(config)# no ip igmp snooping vlan 1 querier
election
```

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 Snooping 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-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

The following example sets the version of the IGMP Snooping Querier VLAN 1 to 3:

```
console(config)# ip igmp snooping vlan 1 querier version 3
```

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-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
```

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

User 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:

```
console# show ip igmp snooping groups
```

Vlan	Group Address	Source Address	Include Ports	Exclude Ports	Comp-Mode
----	-----	-----	-----	-----	-----
1	239.255.255 .250	*	te0/1		v2

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

- *vlan-id*—Specifies the VLAN ID.

Command Mode

User 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 Querier is globally enabled
IGMP snooping querier global address: 194.10.12.56
VLAN 1000
  IGMP Snooping is enabled
  IGMP snooping last immediate leave: enable
  Automatic learning of Multicast router ports is enabled
  IGMP Snooping Querier is enabled
  IGMP Snooping Querier operation state: is running
  IGMP Snooping Querier version: 2
  IGMP Snooping Querier election is enabled
  IGMP Snooping Querier address: 194.12.10.166
  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
  Groups that are in IGMP version 1 compatibility mode:
    231.2.2.3, 231.2.2.3
```

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

- *vlan-id*—Specifies the VLAN ID.

Command Mode

User 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	te0/1	te0/2	te0/3-4

show ip igmp snooping multicast-tv

The `show ip igmp snooping multicast-tv` EXEC mode command displays the IP addresses associated with Multicast TV VLANs.

Syntax

`show ip igmp snooping multicast-tv [vlan vlan-id]`

Parameters

- *vlan-id*—Specifies the VLAN ID.

Command Mode

User EXEC mode

Example

The following example displays the IP addresses associated with all Multicast TV VLANs.

```
console# show ip igmp snooping multicast-tv
VLAN IP Address
----
1000 239.255.0.0
1000 239.255.0.1
1000 239.255.0.2
1000 239.255.0.3
1000 239.255.0.4
1000 239.255.0.5
1000 239.255.0.6
1000 239.255.0.7
```


IGMP Commands

ip igmp last-member-query-count

To configure the Internet Group Management Protocol (IGMP) last member query counter, use the **ip igmp last-member-query-count** command in interface configuration mode. To restore the default value, use the **no** form of this command.

Syntax

```
ip igmp last-member-query-count count  
no ip igmp last-member-query-count
```

Parameters

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 IGMP Robustness variable.

Command Mode

Interface configuration

User Guidelines

Use the **ip igmp robustness** command to change the IGMP last member query counter.

Example

The following example changes a value of the IGMP last member query counter to 3:

```
interface vlan 1  
  ip igmp last-member-query-count 3
```

exit

ip igmp last-member-query-interval

To configure the Internet Group Management Protocol (IGMP) last member query interval, use the `ip igmp last-member-query-interval` command in interface configuration mode. To restore the default IGMP query interval, use the `no` form of this command.

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

The default IGMP last member query interval is 1000 milliseconds.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

Use the `ip igmp last-member-query-interval` command to configure the IGMP last member query interval on an interface.

Example

The following example shows how to increase the the IGMP last member query interval to 1500 milliseconds:

```
console(config)# interface vlan 100  
console(config-if)# ip igmp last-member-query-interval 1500  
console(config-if)# exit
```

ip igmp query-interval

To configure the frequency at which the IGMP querier sends Internet Group Management Protocol (IGMP) host-query messages from an interface, use the **ip igmp query-interval** command in interface configuration mode. To restore the default IGMP query interval, use the **no** form of this command.

Syntax

ip igmp query-interval *seconds*

no ip igmp query-interval

Parameters

- *seconds*—Frequency, in seconds, at which the switch sends IGMP query messages from the interface. The range is from 1 to 31744.

Default Configuration

The default IGMP query interval is 125 seconds.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

Use the **ip igmp query-interval** command to configure the frequency at which the IGMP querier sends IGMP host-query messages from an interface. The IGMP querier sends query-host messages to discover which multicast groups have members on the attached networks of the router.

The query interval must be bigger than the maximum query response time.

Example

The following example shows how to increase the frequency at which the IGMP querier sends IGMP host-query messages to 180 seconds:

```
console(config)# interface vlan 100
console(config-if)# ip igmp query-interval 180
console(config-if)# exit
```

ip igmp query-max-response-time

To configure the maximum response time advertised in Internet Group Management Protocol (IGMP) queries, use the **ip igmp query-max-response-time** command in interface configuration mode. To restore the default value, use the **no** form of this command.

Syntax

ip igmp query-max-response-time *seconds*

no ip query-max-response-time

Parameters

- *seconds*—Maximum response time, in seconds, advertised in IGMP queries. (Range: 0–31744)

Default Configuration

10 seconds.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

This command controls the period during which the responder can respond to an IGMP query message before the router deletes the group.

This command controls how much time the hosts have to answer an IGMP query message before the router deletes their group. Configuring a value of fewer than 10 seconds enables the router to prune groups faster.

The maximum query response time must be less than the query interval.

Note. If the hosts do not respond fast enough, they might be pruned inadvertently. Therefore, the hosts must know to respond faster than 10 seconds (or the value you configure).

Example

The following example configures a maximum response time of 8 seconds:

```
console(config)# interface vlan 100
console(config-if)# ip igmp query-max-response-time 8
console(config-if)# exit
```

ip igmp robustness

To configure the Internet Group Management Protocol (IGMP) robustness variable, use the **ip igmp robustness** command in interface configuration mode. To restore the default value, use the **no** form of this command.

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

The default value is 2.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

Use the **ip igmp robustness** command to change the IGMP robustness variable.

Example

The following example changes a value of the IGMP robustness variable to 3:

```
console(config)# interface vlan 1
```

```
console(config-if)# ip igmp robustness 3  
console(config-if)# exit
```

Link Aggregation Control Protocol Commands

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
```

lacp port-priority

Use the **lacp port-priority** Interface (Ethernet) Configuration 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: 1 use the **no** form of this command 65535)

Default Configuration

The default port priority is 1.

Command Mode

Interface (Ethernet) Configuration mode

Example

The following example sets the priority of te0/1.

```
console(config)# interface te0/1
console(config-if)# lacp port-priority 247
```

lacp timeout

Use the **lacp timeout** Interface (Ethernet) Configuration 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 (Ethernet) Configuration mode

Example

The following example assigns a long administrative LACP timeout to te0/1.

```
console(config)# interface te0/1
console(config-if)# lACP timeout long
```

show lacp

Use the `show lacp` Privileged 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

Privileged EXEC mode

Example

The following example displays LACP information for te0/1.

```
console# show lacp ethernet te0/1

Port te0/1 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 te0/1 LACP Statistics:

```
LACP PDUs sent: 2
LACP PDUs received: 2
```

Port te0/1 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

```

show lacp port-channel

Use the `show lacp port-channel` Privileged 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

Privileged 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 Priority: 1
MAC Address:    000285:0E1C00
Admin Key:     29
Oper Key:      29
```

```
Partner
```

```
System Priority: 0
MAC Address:    00:00:00:00:00:00
Oper Key:      14
```


GARP VLAN Registration Protocol Commands

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
```

gvrp enable (Interface)

Use the `gvrp enable` Interface (Ethernet, Port Channel) Configuration 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 (Ethernet, Port Channel) Configuration 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 #4.

```
console(config)# interface te0/4
console(config-if)# gvrp enable
```

garp timer

Use the `garp timer` Interface Configuration mode command to adjust the values of the join, leave and leaveall timers of GARP applications, such as GVRP. Use the `no` form of this command to restore the default configuration.

Syntax

```
garp timer {join / leave / leaveall} timer-value
```

no garp timer

Parameters

- The following specify the type of timer. The possible values are:
 - **join**—Specifies the GARP join timer. The timer value for this type of timer specifies the time interval between the two join messages sent by the GARP application.
 - **leave**—Specifies the GARP leave timer. The timer value for this type of timer specifies the time interval for a GARP application to wait for a join message after receiving a leave message for a GARP attribute, before it de-registers the GARP attribute.
 - **leaveall**—Specifies the GARP leaveall timer. The timer value for this type of timer specifies the time interval between leaveall messages for a GARP entity, which prompt other GARP entities to re-reregister all attribute information on this entity.
- **timer-value**—Specifies the timer value in milliseconds in multiples of 10. (Range: 10–2147483640)

Default Configuration

The following are the default timer values:

- **Join timer**—200 milliseconds
- **Leave timer**—600 milliseconds
- **Leaveall timer**—10000 milliseconds

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The **timer-value** must be a multiple of 10.

The following relationship must be maintained between the timers:

- The leave timer value must be greater than or equal to three times the join timer.
- The leave-all timer value must be greater than the leave timer.

Set the same GARP timer values on all Layer 2-connected devices to ensure proper operation of the GARP application.

Example

The following example sets the leave timer for #4 to 900 milliseconds.

```
console(config-if)# interface te0/4
console(config-if)# garp timer leave 900
```

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 (Ethernet, Port Channel) Configuration mode

Example

The following example disables dynamic VLAN creation on #3.

```
console(config-if)# interface te0/3
console(config-if)# gvrp vlan-creation-forbid
```

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 (Ethernet, Port Channel) Configuration mode

Example

The following example forbids dynamic registration of VLANs on #2.

```
console(config-if)# interface te0/2
console(config-if)# gvrp registration-forbid
```

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 #4.

```
console#clear gvrp statistics #4
```

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 / 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.

Default Configuration

All GVRP statistics are displayed for all interfaces. If `detailed` is not used, only present ports are displayed.

Command Mode

User 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(s)	GVRP Status	Regist- ration	Dynamic VLAN Creation	Timers(ms)		
				Join	Leave	Leave All
te0/1	Enabled	Forbidden	Disabled	600	200	10000
te0/2	Enabled	Normal	Enabled	1200	400	20000

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

User 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
#1	0	0	0	0	0	0	0	0	0	0	0	0
#2	0	0	0	0	0	0	0	0	0	0	0	0
#3	0	0	0	0	0	0	0	0	0	0	0	0
#4	0	0	0	0	0	0	0	0	0	0	0	0

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

User 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
-----
#1       0        0        0        0        0
#2       0        0        0        0        0
#3       0        0        0        0        0
#4       0        0        0        0        0
```


DHCP Snooping Commands

ip dhcp snooping

Use the **ip dhcp snooping** Global Configuration mode command to enable Dynamic Host Configuration Protocol (DHCP) Snooping globally. Use the **no** form of this command to restore the default configuration.

Syntax

ip dhcp snooping

no ip dhcp snooping

Parameters

N/A

Default Configuration

DHCP snooping is disabled.

Command Mode

Global Configuration mode

User Guidelines

For any DHCP Snooping configuration to take effect, DHCP Snooping must be enabled globally. DHCP Snooping on a VLAN is not active until DHCP Snooping on a VLAN is enabled by using the **ip dhcp snooping vlan** Global Configuration mode command.

Example

The following example enables DHCP Snooping on the device.

```
console(config)# ip dhcp snooping
```

ip dhcp snooping vlan

Use the `ip dhcp snooping vlan` Global Configuration mode command to enable DHCP Snooping on a VLAN. Use the `no` form of this command to disable DHCP Snooping on a VLAN.

Syntax

```
ip dhcp snooping vlan vlan-id
```

```
no ip dhcp snooping vlan vlan-id
```

Parameters

- *vlan-id*—Specifies the VLAN ID.

Default Configuration

DHCP Snooping on a VLAN is disabled.

Command Mode

Global Configuration mode

User Guidelines

DHCP Snooping must be enabled globally before enabling DHCP Snooping on a VLAN.

Example

The following example enables DHCP Snooping on VLAN 21.

```
console(config)# ip dhcp snooping vlan 21
```

ip dhcp snooping trust

Use the `ip dhcp snooping trust` Interface Configuration (Ethernet, Port-channel) mode command to configure a port as trusted for DHCP snooping purposes. Use the `no` form of this command to restore the default configuration.

Syntax

`ip dhcp snooping trust`

`no ip dhcp snooping trust`

Parameters

N/A

Default Configuration

The interface is untrusted.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

Configure as trusted the ports that are connected to a DHCP server or to other switches or routers. Configure the ports that are connected to DHCP clients as untrusted.

Example

The following example configures te0/4 as trusted for DHCP Snooping.

```
console(config)# interface te0/4
console(config-if)# ip dhcp snooping trust
```

ip dhcp snooping information option allowed-untrusted

Use the `ip dhcp snooping information option allowed-untrusted` Global Configuration mode command to allow a device to accept DHCP packets with option-82 information from an untrusted port. Use the `no` form of this command to drop these packets from an untrusted port.

Syntax

`ip dhcp snooping information option allowed-untrusted`

`no ip dhcp snooping information option allowed-untrusted`

Parameters

N/A

Default Configuration

DHCP packets with option-82 information from an untrusted port are discarded.

Command Mode

Global Configuration mode

Example

The following example allows a device to accept DHCP packets with option-82 information from an untrusted port.

```
console(config)# ip dhcp snooping information option allowed-untrusted
```

ip dhcp snooping verify

Use the **ip dhcp snooping verify** Global Configuration mode command to configure a device to verify that the source MAC address in a DHCP packet received on an untrusted port matches the client hardware address. Use the **no** form of this command to disable MAC address verification in a DHCP packet received on an untrusted port.

Syntax

ip dhcp snooping verify

no ip dhcp snooping verify

Default Configuration

The switch verifies that the source MAC address in a DHCP packet received on an untrusted port matches the client hardware address in the packet.

Command Mode

Global Configuration mode

Example

The following example configures a device to verify that the source MAC address in a DHCP packet received on an untrusted port matches the client hardware address.

```
console(config)# ip dhcp snooping verify
```

ip dhcp snooping database

Use the `ip dhcp snooping database` Global Configuration mode command to enable the DHCP Snooping binding database file. Use the `no` form of this command to delete the DHCP Snooping binding database file.

Syntax

`ip dhcp snooping database`

`no ip dhcp snooping database`

Parameters

N/A

Default Configuration

The DHCP Snooping binding database file is not defined.

Command Mode

Global Configuration mode

User Guidelines

The DHCP Snooping binding database file resides on Flash.

To ensure that the lease time in the database is accurate, the Simple Network Time Protocol (SNTP) must be enabled and configured.

The device writes binding changes to the binding database file only if the device system clock is synchronized with SNTP.

Example

The following example enables the DHCP Snooping binding database file.

```
console(config)# ip dhcp snooping database
```

ip dhcp snooping database update-freq

Use the `ip dhcp snooping database update-freq` Global Configuration mode command to set the update frequency of the DHCP Snooping binding database file. Use the `no` form of this command to restore the default configuration.

Syntax

```
ip dhcp snooping database update-freq seconds
```

```
no ip dhcp snooping database update-freq
```

Parameters

- *seconds*—Specifies the update frequency in seconds. (Range: 600–86400).

Default Configuration

The default update frequency value is 1200 seconds.

Command Mode

Global Configuration mode

Example

The following example sets the DHCP Snooping binding database file update frequency to 1 hour.

```
console(config)# ip dhcp snooping database update-freq 3600
```

ip dhcp snooping binding

Use the `ip dhcp snooping binding` Global Configuration mode command to configure the DHCP Snooping binding database and add binding entries to

the database. Use the **no** form of this command to delete entries from the binding database.

Syntax

```
ip dhcp snooping binding mac-address vlan-id ip-address interface-id expiry  
{seconds / infinite}
```

```
no ip dhcp snooping binding mac-address vlan-id
```

Parameters

- *mac-address*—Specifies a MAC address.
- *vlan-id*—Specifies a VLAN number.
- *ip-address*—Specifies an IP address.
- *interface-id*—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.
- **expiry**
 - *seconds*—Specifies the time interval, in seconds, after which the binding entry is no longer valid. (Range: 10–4294967294).
 - **infinite**—Specifies infinite lease time.

Default Configuration

No static binding exists.

Command Mode

Global Configuration mode

User Guidelines

After entering this command, an entry is added to the DHCP Snooping database. If the DHCP Snooping binding file exists, the entry is also added to that file.

The entry is displayed in the show commands as a DHCP Snooping entry.

The user cannot delete dynamic temporary entries for which the IP address is 0.0.0.0.

The user can add static entry to the DHCP Snooping database by using the command **ip source-guard binding**.

Example

The following example adds a binding entry to the DHCP Snooping binding database.

```
console(config)# ip dhcp snooping binding 0060.704C.73FF 23  
176.10.1.1 te0/4 expiry 900
```

clear ip dhcp snooping database

Use the `clear ip dhcp snooping database` Privileged EXEC mode command to clear the DHCP Snooping binding database.

Syntax

`clear ip dhcp snooping database`

Parameters

N/A

Command Mode

Privileged EXEC mode

Example

The following example clears the DHCP Snooping binding database.

```
console# clear ip dhcp snooping database
```

show ip dhcp snooping

Use the `show ip dhcp snooping` EXEC mode command to display the DHCP snooping configuration for all interfaces or for a specific interface.

Syntax

`show ip dhcp snooping [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

User EXEC mode

Example

The following example displays the DHCP snooping configuration.

```
console# show ip dhcp snooping
DHCP snooping is Enabled
DHCP snooping is configured on following VLANs: 21
DHCP snooping database is Enabled
Relay agent Information option 82 is Enabled
Option 82 on untrusted port is allowed
Verification of hwaddr field is Enabled
DHCP snooping file update frequency is configured to: 6666 seconds
```

Interface	Trusted
-----	-----
te0/1	Yes
te0/2	Yes

show ip dhcp snooping binding

Use the `show ip dhcp snooping binding` User EXEC mode command to display the DHCP Snooping binding database and configuration information for all interfaces or for a specific interface.

Syntax

```
show ip dhcp snooping binding [mac-address mac-address] [ip-address ip-address] [vlan vlan-id] [interface-id]
```

Parameters

- **mac-address** *mac-address*—Specifies a MAC address.
- **ip-address** *ip-address*—Specifies an IP address.
- **vlan** *vlan-id*—Specifies a VLAN ID.
- **interface-id**—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

Command Mode

User EXEC mode

Example

The following examples displays the DHCP snooping binding database and configuration information for all interfaces on a device.-

```
console# show ip dhcp snooping binding
```

```
Update frequency: 1200
```

```
Total number of binding: 2
```

Mac Address	IP Address	Lease (sec)	Type	VLAN	Interface
-----	-----	-----	-----	----	-----
0060.704C.73FF	-----	7983	snooping	3	te0/1
0060.704C.7BC1	10.1.8.1	92332	snooping	3	te0/2
	10.1.8.2		(s)		

ip source-guard binding

Use the **ip source-guard binding** Global Configuration mode command to configure the static IP source bindings on the device. Use the **no** form of this command to delete the static bindings.

Syntax

```
ip source-guard binding mac-address vlan-id ip-address interface-id
```

```
no ip source-guard binding mac-address vlan-id
```

Parameters

- *mac-address*—Specifies a MAC address.
- *vlan-id*—Specifies a VLAN number.
- *ip-address*—Specifies an IP address.
- *interface-id*—Specifies an interface ID. The interface ID can be one of the following types: Ethernet port or Port-channel.

Default Configuration

No static binding exists.

Command Mode

Global Configuration mode

User Guidelines

The device currently supports filtering that is based only on the source IP address. In future, the device might support filtering mode that is based on the MAC address and IP source address. Currently the MAC address field is an informative field.

Example

The following example configures the static IP source bindings.

```
console(config)# ip source-guard binding 0060.704C.73FF 23  
176.10.1.1 te0/4
```

ip arp inspection

Use the **ip arp inspection** Global Configuration mode command globally to enable Address Resolution Protocol (ARP) inspection. Use the **no** form of this command to disable ARP inspection.

Syntax

ip arp inspection

no ip arp inspection

Parameters

N/A

Default Configuration

ARP inspection is disabled.

Command Mode

Global Configuration mode

User Guidelines

Note that if a port is configured as an untrusted port, then it should also be configured as an untrusted port for DHCP Snooping, or the IP-address-MAC-address binding for this port should be configured statically. Otherwise, hosts that are attached to this port cannot respond to ARPs.

Example

The following example enables ARP inspection on the device.

```
console(config)# ip arp inspection
```

ip arp inspection vlan

Use the `ip arp inspection vlan` Global Configuration mode command to enable ARP inspection on a VLAN, based on the DHCP Snooping database. Use the `no` form of this command to disable ARP inspection on a VLAN.

Syntax

```
ip arp inspection vlan vlan-id  
no ip arp inspection vlan vlan-id
```

Parameters

- *vlan-id*—Specifies the VLAN ID.

Default Configuration

DHCP Snooping based ARP inspection on a VLAN is disabled.

Command Mode

Global Configuration mode

User Guidelines

This command enables ARP inspection on a VLAN based on the DHCP snooping database. Use the **ip arp inspection list assign** Global Configuration mode command to enable static ARP inspection.

Example

The following example enables DHCP Snooping based ARP inspection on VLAN 23.

```
console(config)# ip arp inspection vlan 23
```

ip arp inspection trust

Use the **ip arp inspection trust** Interface Configuration (Ethernet, Port-channel) mode command to configure an interface trust state that determines if incoming Address Resolution Protocol (ARP) packets are inspected. Use the **no** form of this command to restore the default configuration.

Syntax

ip arp inspection trust

no ip arp inspection trust

Parameters

N/A

Default Configuration

The interface is untrusted.

Command Mode

Interface (Ethernet, Port Channel) Configuration mode

User Guidelines

The device does not check ARP packets that are received on the trusted interface; it only forwards the packets.

For untrusted interfaces, the device intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The device drops invalid packets and logs them in the log buffer according to the logging configuration specified with the `ip arp inspection log-buffer vlan` Global Configuration mode command.

Example

The following example configures `te0/3` as a trusted interface.

```
console(config)# interface te0/3
console(config-if)# ip arp inspection trust
```

ip arp inspection validate

Use the `ip arp inspection validate` Global Configuration mode command to perform specific checks for dynamic Address Resolution Protocol (ARP) inspection. Use the `no` form of this command to restore the default configuration.

Syntax

```
ip arp inspection validate
no ip arp inspection validate
```

Parameters

N/A

Default Configuration

ARP inspection validation is disabled.

Command Mode

Global Configuration mode

User Guidelines

The following checks are performed:

- **Source MAC address:** Compares the source MAC address in the Ethernet header against the sender MAC address in the ARP body. This check is performed on both ARP requests and responses.
- **Destination MAC address:** Compares the destination MAC address in the Ethernet header against the target MAC address in the ARP body. This check is performed for ARP responses.
- **IP addresses:** Compares the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses.

Example

The following example executes ARP inspection validation.

```
console(config)# ip arp inspection validate
```

ip arp inspection list create

Use the `ip arp inspection list create` Global Configuration mode command to create a static ARP binding list and enters the ARP list configuration mode. Use the `no` form of this command to delete the list.

Syntax

```
ip arp inspection list create name
```

```
no ip arp inspection list create name
```

Parameters

- *name*—Specifies the static ARP binding list name. (Length: 1–32 characters).

Default Configuration

No static ARP binding list exists.

Command Mode

Global Configuration mode

User Guidelines

Use the `ip arp inspection list assign` command to assign the list to a VLAN.

Example

The following example creates the static ARP binding list 'servers' and enters the ARP list configuration mode.

```
console(config)# ip arp inspection list create servers
```

ip mac

Use the `ip mac` ARP-list Configuration mode command to create a static ARP binding. Use the `no` form of this command to delete a static ARP binding.

Syntax

```
ip ip-address mac mac-address
```

```
no ip ip-address mac mac-address
```

Parameters

- *ip-address*—Specifies the IP address to be entered to the list.
- *mac-address*—Specifies the MAC address associated with the IP address.

Default Configuration

No static ARP binding is defined.

Command Mode

ARP-list Configuration mode

Example

The following example creates a static ARP binding.

```
console(config)# ip arp inspection list create servers
```

```
console(config-arp-list)# ip 172.16.1.1 mac 0060.704C.7321
console(config-arp-list)# ip 172.16.1.2 mac 0060.704C.7322
```

ip arp inspection list assign

Use the **ip arp inspection list assign** Global Configuration mode command to assign a static ARP binding list to a VLAN. Use the **no** form of this command to delete the assignment.

Syntax

```
ip arp inspection list assign vlan-id name
```

```
no ip arp inspection list assign vlan-id
```

Parameters

- *vlan-id*—Specifies the VLAN ID.
- *name*—Specifies the static ARP binding list name.

Default Configuration

No static ARP binding list assignment exists.

Command Mode

Global Configuration mode

Example

The following example assigns the static ARP binding list Servers to VLAN 37.

```
console(config)# ip arp inspection list assign 37 servers
```

ip arp inspection logging interval

Use the **ip arp inspection logging interval** Global Configuration mode command to set the minimum time interval between successive ARP SYSLOG messages. Use the **no** form of this command to restore the default configuration.

Syntax

`ip arp inspection logging interval {seconds / infinite}`

`no ip arp inspection logging interval`

Parameters

- *seconds*—Specifies the minimum time interval between successive ARP SYSLOG messages. A 0 value means that a system message is immediately generated. (Range: 0–86400)
- *infinite*—Specifies that SYSLOG messages are not generated.

Default Configuration

The default minimum ARP SYSLOG message logging time interval is 5 seconds.

Command Mode

Global Configuration mode

Example

The following example sets the minimum ARP SYSLOG message logging time interval to 60 seconds.

```
console(config)# ip arp inspection logging interval 60
```

show ip arp inspection

Use the `show ip arp inspection EXEC` mode command to display the ARP inspection configuration for all interfaces or for a specific interface.

Syntax

`show ip arp inspection [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

User EXEC mode

Example

The following example displays the ARP inspection configuration.

```
console# show ip arp inspection
IP ARP inspection is Enabled
IP ARP inspection is configured on following VLANs: 1
Verification of packet header is Enabled
IP ARP inspection logging interval is: 222 seconds
  Interface      Trusted
  -----
te0/1            Yes
te0/2            Yes
```

show ip arp inspection list

Use the `show ip arp inspection list` Privileged EXEC mode command to display the static ARP binding list.

Syntax

`show ip arp inspection list`

Parameters

N/A

Command Mode

Privileged EXEC mode

Example

The following example displays the static ARP binding list.

```
console# show ip arp inspection list
List name: servers
Assigned to VLANs: 1,2

IP                ARP
-----          -
172.16.1.1        0060.704C.7322
172.16.1.2        0060.704C.7322
```

show ip arp inspection statistics

Use the `show ip arp inspection statistics EXEC` command to display statistics for the following types of packets that have been processed by this feature: Forwarded, Dropped, IP/MAC Validation Failure.

Syntax

```
show ip arp inspection statistics [vlan vlan-id]
```

Parameters

- *vlan-id*—Specifies VLAN ID.

Command Mode

User EXEC mode

User Guidelines

To clear ARP Inspection counters use the `clear ip arp inspection statistics` CLI command. Counters values are kept when disabling the ARP Inspection feature.

Example

```
console# show ip arp inspection statistics
Vlan   Forwarded Packets Dropped Packets IP/MAC Failures
-----
-----
```

clear ip arp inspection statistics

Use the `clear ip arp inspection statistics` Privileged EXEC mode command to clear statistics ARP Inspection statistics globally.

Syntax

`clear ip arp inspection statistics [vlan vlan-id]`

Parameters

- *vlan-id*—Specifies VLAN ID.

Command Mode

Privileged EXEC mode

Example

```
console# clear ip arp inspection statistics
```


DHCP Relay Commands

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
```

ip dhcp relay enable (Interface)

Use the `ip dhcp relay enable` Interface Configuration 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 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
```

ip dhcp relay address (Global)

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 the server 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

User Guidelines

Use the `ip dhcp relay address` command to define a global DHCP Server IP address. To define a few DHCP Servers, use the command a few times.

To remove a DHCP Server, use the `no` form of the command with the *ip-address* argument.

The `no` form of the command without the *ip-address* argument deletes all global defined DHCP servers.

Example

The following example defines the DHCP server on the device.

```
console(config)# ip dhcp relay address 176.16.1.1
```

ip dhcp relay address (Interface)

Use the `ip dhcp relay address` Interface Configuration (VLAN, Ethernet, Port-channel) command to define the DHCP servers available by the DHCP relay for DHCP clients connected to the interface. Use the `no` form of this command to remove the server 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

Interface Configuration mode

User Guidelines

Use the `ip dhcp relay address` command to define a DHCP Server IP address per the interface. To define multiple DHCP Servers, use the command multiple times.

To remove a DHCP server, use the `no` form of the command with the *ip-address* argument.

The `no` form of the command without the *ip-address* argument deletes all DHCP servers.

Example

The following example defines the DHCP server on the device.

```
console(config-if)# ip dhcp relay address 176.16.1.1
```

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

User EXEC mode

Example

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: te0/1,pol-2
  Active:
  Inactive: te0/1, pol-4
DHCP relay is enabled on VLANs: 1, 2, 4, 5
  Active:
  Inactive: 1, 2, 4, 5
Global 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: te0/1,pol-2
  Active: te0/1
  Inactive: pol-2
DHCP relay is enabled on VLANs: 1, 2, 4, 5
  Active: 1, 2, 4, 5
  Inactive:
Global Servers: 1.1.1.1 , 2.2.2.2
```

Example 3. Option 82 is supported (enabled) and there DHCP Servers defined per interface:

```
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: te0/1,pol-2
  Active: te0/1
  Inactive: pol-2
DHCP relay is enabled on VLANs: 1, 2, 4, 5
  Active: 1, 2, 4, 5
  Inactive:
Global Servers: 1.1.1.1 , 2.2.2.2
VLAN 1: 1.1.1.1, 100.10.1.1
VLAN 2: 3.3.3.3, 4.4.4.4, 5.5.5.5
VLAN 10: 6.6.6.6
```

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
```

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

User EXEC mode

Example

The following example displays the DHCP Option 82 configuration.

```
console# show ip dhcp information option
```

```
Relay agent Information option is Enabled
```


DHCPv6 Commands

ipv6 dhcp client stateless

Use the `ipv6 dhcp client stateless` command in Interface Configuration mode to enable DHCP for an IPv6 client process and to enable request for stateless configuration through the interface on which the command is run. To disable requests for stateless configuration, use the `no` form of this command.

Syntax

`ipv6 dhcp client stateless`

`no ipv6 dhcp client stateless`

Parameters

This command has no arguments or keywords.

Default Configuration

Information request is disabled on an interface.

Command Mode

Global Configuration mode

User Guidelines

Enabling this command starts the DHCPv6 client process if this process is not yet running and IPv6 interface is enabled on the interface.

This command enables the DHCPv6 Stateless service on the interface. The service allows to receive the configuration from a DHCP server, passed in the following options:

- Option 23: `OPTION_DNS_SERVERS` - List of DNS Servers IPv6 Addresses
- Option 24: `OPTION_DOMAIN_LIST` - Domain Search List
- Option 31: `OPTION_SNTP_SERVERS` - List of SNTP Servers IPv6 Addresses

- Option 32: OPTION_INFORMATION_REFRESH_TIME - Information Refresh Time Option
- Option 41: OPTION_NEW_POSIX_TIMEZONE - New Timezone Posix String
- Option 59: OPT_BOOTFILE_URL - Configuration Server URL
- Option 60: OPT_BOOTFILE_PARAM, the first parameter - Configuration File Path Name

Example

The following example enables the Stateless service:

```
console(config)# interface vlan 100
console(config-if)# ipv6 dhcp client stateless
console(config-if)# exit
```

clear ipv6 dhcp client

Use the `clear ipv6 dhcp client` command in Privileged EXEC mode to restart DHCP for an IPv6 client on an interface.

Syntax

```
clear ipv6 dhcp client interface-id
```

Parameters

- *interface-id*—Interface identifier.

Default Configuration

N/A

Command Mode

Privileged EXEC mode

User Guidelines

This command restarts DHCP for an IPv6 client on a specified interface after first releasing and unconfiguring previously-acquired prefixes and other configuration options (for example, Domain Name System [DNS] servers).

Example

The following example restarts the DHCP for IPv6 client on VLAN 100:

```
console# clear ipv6 dhcp client vlan 100
```

ipv6 dhcp client information refresh

To configure the refresh time for IPv6 client information refresh time on a specified interface if the DHCPv6 server reply does not include the Information Refresh Time, use the **ipv6 dhcp client information refresh** command in Interface Configuration mode. To return to the default value of the refresh time, use the **no** form of this command.

Syntax

ipv6 dhcp client information refresh *seconds* / *infinite*

no ipv6 dhcp client information refresh

Parameters

- *seconds*—The refresh time, in seconds. The value cannot be less than the minimal acceptable refresh time configured by the **ipv6 dhcp client information refresh** command. The maximum value that can be used is 4,294,967,294 seconds (0xFFFFFFFF).
- *infinite*—Infinite refresh time.

Default Configuration

The default is 86,400 seconds (24 hours).

Command Mode

Interface Configuration mode

User Guidelines

The `ipv6 dhcp client information refresh` command specifies the information refresh time. If the server does not send an information refresh time option then a value configured by the command is used.

Use the `infinite` keyword, to prevent refresh, if the server does not send an information refresh time option.

Example

The following example configures an upper limit of 2 days:

```
console(config)# interface vlan 100
console(config-if)# ipv6 dhcp client stateless
console(config-if)# ipv6 dhcp client information refresh 172800
console(config-if)# exit
```

ipv6 dhcp client information refresh minimum

To configure the minimum acceptable refresh time on the specified interface, use the `ipv6 dhcp client information refresh minimum` command in Interface Configuration mode. To remove the configured refresh time, use the `no` form of this command.

Syntax

`ipv6 dhcp client information refresh minimum seconds / infinite`

`no ipv6 dhcp client information refresh minimum`

Parameters

- *seconds*—The refresh time, in seconds. The minimum value that can be used is 600 seconds. The maximum value that can be used is 4,294,967,294 seconds (0xFFFFFFFF).
- *infinite*—Infinite refresh time.

Default Configuration

The default is 86,400 seconds (24 hours).

Command Mode

Interface Configuration mode

User Guidelines

The `ipv6 dhcp client information refresh minimum` command specifies the minimum acceptable information refresh time. If the server sends an information refresh time option of less than the configured minimum refresh time, the configured minimum refresh time will be used instead.

This command may be configured in the following situations:

- In unstable environments where unexpected changes are likely to occur.
- For planned changes, including renumbering. An administrator can gradually decrease the time as the planned event nears.
- Limit the amount of time before new services or servers are available to the client, such as the addition of a new Simple Network Time Protocol (SNTP) server or a change of address of a Domain Name System (DNS) server.

If you configure the `infinite` keyword client never refreshes the information.

Example

The following example configures an upper limit of 2 days:

```
console(config)# interface vlan 100
console(config-if)# ipv6 dhcp client stateless
console(config-if)# ipv6 dhcp client information refresh 172800
console(config-if)# exit
```

ipv6 dhcp duid-en

Use the `ipv6 dhcp duid-en` command in Global Configuration mode to set the Vendor Based on Enterprise Number DHCPv6 Unique Identified (DUID-EN) format.

To return to the default value, use the `no` form of this command.

Syntax

`ipv6 dhcp duid-en enterprise-number identifier`

no ipv6 dhcp duid-en

Parameters

- *enterprise-number*—The vendor’s registered Private Enterprise number as maintained by IANA.
- *identifier*—The vendor-defined non-empty hex string (up to 64 hex characters). If the number of the character is not even '0' is added at the right. Each 2 hex characters can be separated by a period or colon.

Default Configuration

DUID Based on Link-layer Address (DUID-LL) is used. The base MAC Address is used as a Link-layer Address.

Command Mode

Global Configuration mode

User Guidelines

By default, the DHCPv6 uses the DUID Based on Link-layer Address (see RFC3315) with the Base MAC Address as a Link-layer Address.

Use this command to change the UDID format to the Vendor Based on Enterprise Number.

Example

Example 1. The following sets the DIID-EN format:

```
console(config)# ipv6 dhcp udid-en 9 0CC084D303000912
```

Example 2. The following sets the DIID-EN format using colons as delimiter:

```
console(config)# ipv6 dhcp udid-en 9 0C:C0:84:D3:03:00:09:12
```

show ipv6 dhcp

Use the `show ipv6 dhcp` command in User EXEC or Privileged EXEC mode to display the Dynamic DHCP unique identifier (DUID) on a specified device. This information is relevant for DHCPv6 clients and DHCPv6 relays.

Syntax

`show ipv6 dhcp`

Parameters

NA

Command Mode

User EXEC mode

Privileged EXEC mode

User Guidelines

This command uses the DUID, which is based on the link-layer address for both client and server identifiers. The device uses the MAC address from the lowest-numbered interface to form the DUID. The network interface is assumed to be permanently attached to the device.

Example

Example 1. The following is sample output from this command when the switch's UDID format is vendor based on enterprise number:

```
console# show ipv6 dhcp
The switch's DHCPv6 unique identifier(DUID)is
000200000090CC084D303000912
  Format: 2
  Enterprise Number: 9
  Identifier: 0CC084D303000912
```

Example 2. The following is sample output from this command when the switch's UDID format is the vendor-based on link-layer address:

```
console# show ipv6 dhcp
The switch's DHCPv6 unique identifier(DUID)is 000300010024012607AA
  Format: 3
  Hardware type: 1
  MAC Address: 0024.0126.07AA
```

show ipv6 dhcp interface

Use the `show ipv6 dhcp interface` command in User EXEC or Privileged EXEC mode to display DHCP for IPv6 interface information.

Syntax

```
show ipv6 dhcp interface [interface-id]
```

Parameters

- *interface-id*—Interface identifier.

Command Mode

User EXEC mode

Privileged EXEC mode

User Guidelines

If no interfaces are specified in the command, all interfaces on which DHCP for IPv6 (client or server) is enabled are displayed. If an interface is specified in the command, only information about the specified interface is displayed.

Example

Example 1. The following is sample output from this command when only the Stateless service is enabled:

```
console# show ipv6 dhcp interface
VLAN 100 is in client mode
```



```
DHCP Operational mode is enabled
Stateless Service is enabled
Reconfigure service is enabled
Information Refresh Minimum Time: 600 seconds
Information Refresh Time: 86400 seconds
Received Information Refresh Time: 3600 seconds
Remain Information Refresh Time: 411 seconds
DHCP server:
  Address FE80::202:FCFF:FEA1:7439, DUID 000300010002FCA17400
Preference: 20
DNS Servers: 1001::1, 2001::10
DNS Domain Search List: company.com beta.org
SNTP Servers: 2004::1
POSIX Timezone string: EST5EDT4,M3.2.0/02:00,M11.1.0/02:00
Configuration Server: config.company.com
Configuration Path Name: qqg/config/aaa_config.dat
Indirect Image Path Name: qqg/config/aaa_image_name.txt
VLAN 110 is in client mode
  DHCP Operational mode is disabled (IPv6 is not enabled)
  Stateless Service is enabled
  Reconfigure service is enabled
  Information Refresh Minimum Time: 600 seconds
  Information Refresh Time: 86400 seconds
  Remain Information Refresh Time: 0 seconds
VLAN 1000 is in client mode
  DHCP Operational mode is disabled (Interface status is DOWN)
  Stateless Service is enabled
  Reconfigure service is enabled
  Information Refresh Minimum Time: 600 seconds
  Information Refresh Time: 86400 seconds
  Remain Information Refresh Time: 0 seconds
VLAN 2000 is in client mode
  DHCP Operational mode is disabled (Interface status is DOWN)
```

Stateless Service is enabled
Reconfigure service is enabled
Information Refresh Minimum Time: 600 seconds
Information Refresh Time: 86400 seconds
Remain Information Refresh Time: 0 seconds
DHCP server:
 Address FE80::202:FCFF:FEA1:7439, DUID 000300010002FCA17400
Preference: 20
Received Information Refresh Time: 3600 seconds
DNS Servers: 1001::1, 2001::10
DNS Domain Search List: company.com beta.org
SNTP Servers: 2004::1
POSIX Timezone string: EST5EDT4,M3.2.0/02:00,M11.1.0/02:00
Configuration Server: config.company.com
Configuration Path Name: qqq/config/aaa_config.dat
Indirect Image Path Name: qqq/config/aaa_image_name.txt

IP Addressing Commands

IP addresses and Layer 2 Interfaces

IP addresses can be configured on the following Layer 2 interfaces:

- Ethernet port
- Port channel
- VLAN
- OOB port

Lists of Commands

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

OOB port:

```
ip address ip-address {mask | /prefix-length} [default-gateway ip-address]
```

```
no ip address
```

In-Band interfaces:

```
ip address ip-address {mask | /prefix-length}
```

```
no ip address [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. The route is gotten a metric of 1 for an In-Band interface and 1 for OOB.

Default Configuration

No IP address is defined for interfaces.

Command Mode

Interface Configuration mode

User Guidelines

Use the **ip address** command to defines a static IP address on an interface.

In-Band interfaces

Multiple IP addresses are supported. A new defined IP address is added on the interface.

If a configured IP address overlaps another configured one a warning message is displayed. To change an existed IP address, delete the existed one and add the new one.

OOB port

One IP address is supported. A new IP address defined on the OOB port overrides the previously defined IP address on the OOB port.

Defining a static IP address on the OOB port stops a DHCP client running on the OOB port and deletes an IP address assigned by the DHCP client.

While no IP address is assigned either by DHCP client or manually the default IP address **192.168.2.1** is assigned on the OOB port

Example

Example 1. 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
```

Example 2. The following example configures 3 overlapped IP addresses.

```
console(config)# interface vlan 1
console(config-if)# ip address 1.1.1.1 255.0.0.0
console(config)# exit
console(config)# interface vlan 2
console(config-if)# ip address 1.2.1.1 255.255.0.0
console(config)# This IP address overlaps IP address
1.1.1.1/8 on vlan1, are you sure? [Y/N]Y
console(config)# exit
console(config)# interface vlan 3
console(config-if)# ip address 1.3.1.1 255.255.0.0
console(config)# This IP address overlaps IP address
1.1.1.1/8 on vlan1, are you sure? [Y/N]Y
console(config)# exit
```

Example 3. The following example configures IP address on OOB:

```
console(config)# interface oob
console(config-if)# ip address 131.108.1.27
255.255.255.0 default-gateway 131.108.1.100
```

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

N/A

Command Mode

Interface Configuration mode

User Guidelines

Use the **ip address dhcp** command to enable DHCP client on the interface.

The **ip address dhcp** command removes all the manually configured addresses on the interface.

The default route (Default Gateway) received in DHCP Router option (Option 3) is assigned a metric of 1 for an In-Band interface and 253 for OOB.

Use the **no** form of the command to disable DHCP client on interface.

Example

The following example acquires an IP address for VLAN 100 from DHCP.

```
console(config)# interface vlan100  
console(config-if)# ip address dhcp
```

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*—Specifies an interface.
- **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

Use the **renew dhcp** command to renew a DHCP address on an interface.

This command does not enable DHCP client on an interface and if DHCP client is not enabled on the interface, the command returns an error message.

Example

The following example renews an IP address on VLAN 19 that was acquired from a DHCP server:

```
console# renew dhcp vlan 19
```

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 [ip-address]
```

Parameters

- *ip-address*—Specifies the default gateway IP address.

Command Mode

Global Configuration mode

Default Configuration

No default gateway is defined.

User Guidelines

Use the **ip default-gateway** command to defines a default gateway (default route).

The **ip default-gateway** command adds the default route with metric of 1 for the gateway connected on an In-Band interface and 1 for the gateway connected on OOB.

Use the **no ip default-gateway *ip-address*** command to delete one default gateway.

Use the **no ip default-gateway** command to delete all default gateways.

Example

The following example defines default gateway 192.168.1.1.

```
console(config)# ip default-gateway 192.168.1.1
```

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 on which IP addresses are defined.

Default Configuration

All IP addresses.

Command Mode

User EXEC mode

Examples

Example 1 - The following example displays all configured IP addresses and their types:

```

console# show ip interface
!source_precedence_is_supported &&
!broadcast_address_configuration_is_supported &&
!ip_redirects_is_supported
IP Address      I/F      I/F Status  Type   Directed  Status
                admin/oper          Broadcast
-----
10.5.230.232/24  vlan 1   UP/UP      Static  disable  Valid
10.5.234.202/24  vlan 4   UP/DOWN    Static  disable  Valid
10.5.240.200/24  oob      UP/UP      Static          Valid

```

Example 2 - The following example displays the IP addresses configured on the given L2 interfaces and their types:

```

console# show ip interface vlan 1
!source_precedence_is_supported &&
!broadcast_address_configuration_is_supported &&
!ip_redirects_is_supported
IP Address      I/F      I/F Status  Type   Directed  Status
                admin/oper          Broadcast
-----
10.5.230.232/24  vlan 1   UP/UP      Static  disable  Valid

```

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*—Address pair is added for specified interface.

Command Mode

Global Configuration mode

Default Configuration

No permanent entry is defined.

If no interface ID is entered, address pair is relevant to all interfaces.

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 vlan100
```

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, if IP Routing is enabled, and 300 seconds if IP Routing is disabled.

Command Mode

Global Configuration mode

Example

The following example configures the ARP timeout to 12000 seconds.

```
console(config)# arp timeout 12000
```

arp timeout(Interface)

Use the `arp timeout` interface Configuration command to configure how long an entry remains in the ARP cache for specific interface. Use the `no` form of this command restore the default value.

Syntax

`arp timeout seconds`

`no arp timeout`

Parameters

- *seconds*—Time (in seconds) that an entry remains in the ARP cache. It is recommended not to set it to less than 3600. (Range: 1–40000000).

Default

Defined by the `arp timeout` Global Configuration command

Command Mode

Interface Configuration mode

User Guidelines

This configuration can be applied only if at least one IP address is defined on specific interface.

Example

```
console(config)# interface vlan 1
console(config-if)# arp timeout 12000
```

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
```

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.

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           HW           Time-      Staus
-----      -
VLAN 1    te0/1          10.7.1.102   00:10:B5:04:DB:4B   100        Dynamic
VLAN 1    te0/2          10.7.1.135   00:50:22:00:2A:A4   Static
VLAN 2    te0/1          11.7.1.135   00:12:22:00:2A:A4   1000       Dynamic
          te0/2          12.10.1.13   00:11:55:04:DB:4B   Dynamic
```

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

This command has no arguments or key words.

Command Mode

Privileged EXEC mode

Example

```
console# show arp configuration
Global configuration:
```

```
    ARP timeout:      80000 Seconds
Interface configuration:
VLAN 1:
    ARP timeout:60000 Seconds
VLAN 10:
    ARP timeout:70000 Seconds
VLAN 20:
    ARP timeout:80000 Second (Global)
```

ip helper-address

Use the `ip helper-address` Global Configuration mode command to enable the forwarding of 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). This can be a list of port numbers separated by spaces.

Default Configuration

Forwarding of 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

This command forwards specific UDP Broadcast packets from one interface to another, by specifying 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)

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-interface* argument cannot be the OOB port.

Example

The following example enables the forwarding of UDP Broadcast packets received on all interfaces to the UDP ports of a destination IP address and UDP port 1 and 2.

```
console(config)#ip helper-address all 172.16.9.9 49 53 1 2
```

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

This command has no arguments or key words.

Command Mode

Privileged EXEC mode

User Guidelines**Example**

The following example displays the IP helper addresses configuration on the system:

```
console# show ip
```

Interface	Helper Address	UDP Ports
-----	-----	-----
192.168.1.1	172.16.8.8	37, 42, 49, 53, 137
192.168.2.1	172.16.9.9	37, 49

show ip dhcp client interface

Use the `show ip dhcp client interface` command in User EXEC or Privileged EXEC mode to display DHCP client interface information.

Syntax

show ip dhcp client interface [*interface-id*]

Parameters

- *interface-id*—Interface identifier.

Command Mode

User EXEC mode

User Guidelines

If no interfaces are specified, all interfaces on which DHCP client is enabled are displayed. If an interface is specified, only information about the specified interface is displayed.

Example

The following is sample output of the `show ip dhcp client interface` command:

```
console# show ip dhcp client interface
VLAN 100 is in client mode
  Address: 170.10.100.100 Mask: 255.255.255.0 T1 120, T2 192
  Default Gateway: 170.10.100.1
  DNS Servers: 115.1.1.1, 87.12.34.20
  DNS Domain Search List: company.com
  Host Name: switch_floor7
  Configuration Server Addresses: 192.1.1.1 202.1.1.1
  Configuration Path Name: qqq/config/aaa_config.dat
  POSIX Timezone string: EST5EDT4,M3.2.0/02:00,M11.1.0/02:00
VLAN 1200 is in client mode
  Address: 180.10.100.100 Mask: 255.255.255.0 T1 120, T2 192
  Default Gateway: 180.10.100.1
  DNS Servers: 115.1.1.1, 87.12.34.20
  DNS Domain Search List: company.com
  Host Name: switch_floor7
  Configuration Server Addresses: configuration.company.com
  Configuration Path Name: qqq/config/aaa_config.dat
  POSIX Timezone string: EST5EDT4,M3.2.0/02:00,M11.1.0/02:00
```


IPv6 Commands

clear ipv6 neighbors

Use the `clear ipv6 neighbors` command in privileged EXEC mode to delete all entries in the IPv6 neighbor discovery cache, except static entries.

Syntax

```
clear ipv6 neighbors
```

Parameters

N/A

Command Mode

Privileged EXEC mode

User Guidelines

Example

The following example deletes all entries, except static entries, in the neighbor discovery cache:

```
console# clear ipv6 neighbors
```

ipv6 address

Use the `ipv6 address` command in Interface Configuration mode to configure an IPv6 address based on an IPv6 general prefix and enable IPv6 processing on an interface. To remove the address from the interface, use the **no** form of this command.

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 RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.
- *prefix-length*—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 value.

Default Configuration

No IP address is defined for the interface.

Command Mode

Interface Configuration mode

User Guidelines

The `ipv6 address` command cannot be applied to define an IPv6 address on an ISATAP interface.

Using the `no IPv6 address` command without arguments removes all manually-configured IPv6 addresses from an interface, including link local manually configured addresses.

Example

The following example defines the IPv6 global address 2001:DB8:2222:7272::72 on vlan 100:

```
console(config)# interface vlan 100
console(config-if)# ipv6 address 2001:DB8:2222:7272::72/64
console(config-if)# exit
```

ipv6 address anycast

Use the `ipv6 address anycast` command in Interface Configuration mode to configure an IPv6 Anycast address and enable IPv6 processing on an interface.

To remove the address from the interface, use the **no** form of this command.

Syntax

```
ipv6 address ipv6-prefix/prefix-length anycast  
no ipv6 address [ipv6-prefix/prefix-length]
```

Parameters

- *ipv6-address*—Specifies the IPv6 network assigned to the interface. This argument must be in the form documented in RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.
- *prefix-length*—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 value.

Default Configuration

No IP address is defined for the interface.

Command Mode

Interface Configuration mode

User Guidelines

An Anycast address is an address that is assigned to a set of interfaces that typically belong to different nodes. A packet sent to an Anycast address is delivered to the closest interface—as defined by the routing protocols in use—identified by the Anycast address. Anycast addresses are syntactically indistinguishable from Unicast addresses because Anycast addresses are allocated from the Unicast address space. Nodes to which the Anycast address is assigned must be explicitly configured to recognize that the address is an Anycast address.

Anycast addresses can be used only by a router, not a host, and Anycast addresses must not be used as the source address of an IPv6 packet.

The subnet router Anycast address has a prefix concatenated by a series of zeros (the interface ID). The subnet router Anycast address can be used to reach a router on the link that is identified by the prefix in the subnet router Anycast address.

The `ipv6 address anycast` command cannot be applied to define an IPv6 address on an ISATAP interface.

Using the `no IPv6 address` command without arguments removes all manually-configured IPv6 addresses from an interface, including link local manually configured addresses.

Example

The following example enables IPv6 processing on the interface, assigns the prefix `2001:0DB8:1:1::/64` to the interface, and configures the IPv6 Anycast address `2001:0DB8:1:1:FFFF:FFFF:FFFF:FFFE`:

```
console(config)# interface vlan 1
console(config-if)# ipv6 address
2001:0DB8:1:1:FFFF:FFFF:FFFF:FFFE/64 anycast
console(config-if)# exit
```

ipv6 address autoconfig

Use the `ipv6 address autoconfig` command in Interface Configuration mode to enable automatic configuration of IPv6 addresses using stateless auto configuration on an interface and enable IPv6 processing on the interface. Addresses are configured depending on the prefixes received in Router Advertisement messages.

To remove the address from the interface, use the `no` form of this command.

Syntax

```
ipv6 address autoconfig
no ipv6 address autoconfig
```

Parameters

N/A.

Default Configuration

Stateless Auto configuration is disabled.

Command Mode

Interface Configuration mode

User Guidelines

This command enables IPv6 on an interface (if it was disabled) and causes the switch to perform IPv6 stateless address auto-configuration to discover prefixes on the link and then to add the eui-64 based addresses to the interface.

Stateless auto configuration is applied only when IPv6 Forwarding is disabled.

When IPv6 forwarding is changed from disabled to enabled, and stateless auto configuration is enabled the switch stops stateless auto configuration and removes all stateless auto configured ipv6 addresses from all interfaces.

When IPv6 forwarding is changed from enabled to disabled and stateless auto configuration is enabled the switch resumes stateless auto configuration.

Using the **no ipv6 address autoconfig** command to disable stateless auto configuration and to remove all stateless auto configured IPv6 addresses from an interface.

Example

The following example assigns the IPv6 address automatically:

```
console(config)# interface vlan 100
console(config-if)# ipv6 address autoconfig
console(config-if)# exit
```

ipv6 address eui-64

Use the **ipv6 address eui-64** command in Interface Configuration mode to configure an IPv6 address for an interface and enables IPv6 processing on the interface using an EUI-64 interface ID in the low order 64 bits of the address.

To remove the address from the interface, use the **no** form of this command.

Syntax

ipv6 address *ipv6-prefix/prefix-length* **eui-64**

no ipv6 address [*ipv6-prefix/prefix-length* **eui-64**]

Parameters

- *ipv6-address*—Specifies the IPv6 network assigned to the interface. This argument must be in the form documented in RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.
- *prefix-length*—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 value.

Default Configuration

No IP address is defined for the interface.

Command Mode

Interface Configuration mode

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.

If the switch detects another host using one of its IPv6 addresses, it adds the IPv6 address and displays an error message on the console.

Using the **no IPv6 address** command without arguments removes all manually-configured IPv6 addresses from an interface, including link local manually-configured addresses.

Example

The following example enables IPv6 processing on VLAN 1, configures IPv6 global address 2001:0DB8:0:1::/64 and specifies an EUI-64 interface ID in the low order 64 bits of the address:

```
console(config)# interface vlan 1
console(config-if)# ipv6 address 2001:0DB8:0:1::/64 eui-64
console(config-if)# exit
```

ipv6 address link-local

Use the **ipv6 address link-local** command in Interface Configuration mode to configure an IPv6 link local address for an interface and enable IPv6 processing on the interface.

To remove the manually configured link local address from the interface, use the **no** form of this command.

Syntax

```
ipv6 address ipv6-prefix link-local
```

```
no ipv6 address [link-local]
```

Parameters

- *ipv6-address*—Specifies the IPv6 network assigned to the interface. This argument must be in the form documented in RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.

Default Configuration

The default Link-local address is defined.

Command Mode

Interface Configuration mode

User Guidelines

The switch automatically generates a link local address for an interface when IPv6 processing is enabled on the interface, typically when an IPv6 address is configured on the interface. To manually specify a link local address to be used by an interface, use the **ipv6 address link-local** command.

The **ipv6 address link-local** command cannot be applied to define an IPv6 address on an ISATAP interface.

Using the **no IPv6 address** command without arguments removes all manually-configured IPv6 addresses from an interface, including link local manually configured addresses.

Example

The following example enables IPv6 processing on VLAN 1 and configures FE80::260:3EFF:FE11:6770 as the link local address for VLAN 1:

```
console(config)# interface vlan 1
console(config-if)# ipv6 address FE80::260:3EFF:FE11:6770 link-
local
console(config-if)# exit
```

ipv6 default-gateway

Use the `ipv6 default-gateway` Global Configuration mode command to define an IPv6 default gateway. To remove the IPv6 default gateway, use the `no` form of this command.

Syntax

```
ipv6 default-gateway ipv6-address
no ipv6 default-gateway ipv6-address
```

Parameters

- *ipv6-address*—Specifies the IPv6 address of an IPv6 router that can be used to reach a network.

Default Configuration

No default gateway is defined.

Command Mode

Global Configuration mode

User Guidelines

The command is an alias of the `ipv6 route` command with the predefined (default) route:

```
ipv6 route ::/0 ipv6-address | interface-id
```

See the definition of the [ipv6 route](#) command for details.

Examples

Example 1. The following example defines a default gateway with a global IPv6 address:

```
console(config)# ipv6 default-gateway 5::5
```

Example 2. The following example defines a default gateway with a link-local IPv6 address:

```
console(config)# ipv6 default-gateway  
FE80::260:3EFF:FE11:6770%vlan1
```

ipv6 enable

Use the **ipv6 enable** command in Interface Configuration mode to enable IPv6 processing on an interface.

To disable IPv6 processing on an interface that has not been configured with an explicit IPv6 address, use the **no** form of this command.

Syntax

ipv6 enable

no ipv6 enable

Parameters

N/A.

Default Configuration

IPv6 addressing is disabled.

Command Mode

Interface Configuration mode

User Guidelines

This 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 does not disable IPv6 processing on an interface that is configured with an explicit IPv6 address.

Example

The following example enables VLAN 1 for the IPv6 addressing mode.

```
console(config)# interface vlan 1
console(config-if)# ipv6 enable
console(config-if)# exit
```

ipv6 hop-limit

Use the **ipv6 hop-limit** command in Global Configuration mode to configure the maximum number of hops used in all IPv6 packets that are originated by the router.

To return the hop limit to its default value, use the **no** form of this command.

Syntax

ipv6 hop-limit *value*

no ipv6 hop-limit

Parameters

- *value*—Maximum number of hops. The acceptable range is from 1 to 255.

Default Configuration

The default is 64 hops.

Command Mode

Global Configuration mode

Example

The following example configures a maximum number of 15 hops for all IPv6 packets that are originated from the router:

```
console(config)# ipv6 hop-limit 15
```

ipv6 icmp error-interval

Use the `ipv6 icmp error-interval` command in Global Configuration mode to configure the interval and bucket size for IPv6 ICMP error messages. To return the interval to its default setting, use the `no` form of this command.

Syntax

```
ipv6 icmp error-interval milliseconds [bucketsize]
```

```
no ipv6 icmp error-interval
```

Parameters

- *milliseconds*—Time interval between tokens being placed in the bucket. Each token represents a single ICMP error message. The acceptable range is from 0 to 2147483647. A value of 0 disables ICMP rate limiting.
- *bucketsize*—Maximum number of tokens stored in the bucket. The acceptable range is from 1 to 200.

Default Configuration

The default interval is 100ms and the default bucket size is 10 i.e. 100 ICMP error messages per second.

Command Mode

Global Configuration mode

User Guidelines

Use this command to limit the rate at which IPv6 ICMP error messages are sent. A token bucket algorithm is used with one token representing one IPv6 ICMP error message. Tokens are placed in the virtual bucket at a specified interval until the maximum number of tokens allowed in the bucket is reached.

The *milliseconds* argument specifies the time interval between tokens arriving in the bucket. The optional *bucketsize* argument is used to define the maximum number of tokens allowed in the bucket. Tokens are removed from the bucket when IPv6 ICMP error messages are sent, which means that if the *bucketsize* is set to 20, a rapid succession of 20 IPv6 ICMP error messages can

be sent. When the bucket is empty of tokens, IPv6 ICMP error messages are not sent until a new token is placed in the bucket.

Average Packets Per Second = $(1000 / \textit{milliseconds}) * \textit{bucketsize}$.

To disable ICMP rate limiting, set the *milliseconds* argument to zero.

Example

The following example shows an interval of 50 milliseconds and a bucket size of 20 tokens being configured for IPv6 ICMP error messages:

```
console(config)# ipv6 icmp error-interval 50 20
```

ipv6 link-local default zone

Use the **Ipv6 link-local default zone** command to configure an interface to egress a link local packet without a specified interface or with the default zone 0.

Use the **no** form of this command to return the default link local interface to the default value.

Syntax

Ipv6 link-local default zone *interface-id*

no Ipv6 link-local default zone

Parameters

- *interface-id*—Specifies the interface that is used as the egress interface for packets sent without a specified IPv6Z interface identifier or with the default 0 identifier.

Default

By default, **link local default zone** is disabled.

Command Mode

Global Configuration mode

Example

The following example defines VLAN 1 as a default zone:

```
console(config)# ipv6 link-local default zone vlan1
```

ipv6 nd advertisement-interval

Use the `ipv6 nd advertisement-interval` in Interface Configuration mode to configure the advertisement interval option in router advertisements (RAs).

To reset the interval to the default value, use the `no` form of this command.

Syntax

```
ipv6 nd advertisement-interval
```

```
no ipv6 nd advertisement-interval
```

Parameters

N/A.

Default Configuration

Advertisement interval option is not sent.

Command Mode

Interface Configuration mode

User Guidelines

Use the `ipv6 nd advertisement-interval` command to indicate to a visiting mobile node the interval at which that node may expect to receive RAs. The node may use this information in its movement detection algorithm.

Example

The following example enables the advertisement interval option to be sent in RAs:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd advertisement-interval
console(config-if)# exit
```

ipv6 nd dad attempts

Use the `ipv6 nd dad attempts` command in Interface Configuration mode to configure the number of consecutive neighbor solicitation messages that are sent on an interface while duplicate address detection is performed on the Unicast IPv6 addresses of the interface.

To return the number of messages to the default value, use the `no` form of this command.

Syntax

```
ipv6 nd dad attempts value
```

```
no ipv6 nd dad attempts
```

Parameters

- *value*—The number of neighbor solicitation messages. The acceptable range is from 0 to 600. Configuring a value of 0 disables duplicate address detection processing on the specified interface; a value of 1 configures a single transmission without follow-up transmissions.

Default Configuration

```
1
```

Command Mode

Interface Configuration mode

User Guidelines

Duplicate address detection verifies the uniqueness of new Unicast IPv6 addresses before the addresses are assigned to interfaces (the new addresses remain in a tentative state while duplicate address detection is performed). Duplicate address detection uses neighbor solicitation messages to verify the uniqueness of Unicast IPv6 addresses.

The `DupAddrDetectTransmits` node configuration variable (as specified in RFC 4862, IPv6 Stateless Address Autoconfiguration) is used to automatically determine the number of consecutive neighbor solicitation messages that are sent on an interface, while duplicate address detection is performed on a tentative Unicast IPv6 address.

The interval between duplicate address detection, neighbor solicitation messages (the duplicate address detection timeout interval) is specified by the neighbor discovery-related variable `RetransTimer` (as specified in RFC 4861, Neighbor Discovery for IPv6), which is used to determine the time between retransmissions of neighbor solicitation messages to a neighbor when resolving the address or when probing the reachability of a neighbor. This is the same management variable used to specify the interval for neighbor solicitation messages during address resolution and neighbor unreachability detection. Use the `ipv6 nd ns-interval` command to configure the interval between neighbor solicitation messages that are sent during duplicate address detection.

Duplicate address detection is suspended on interfaces that are administratively down. While an interface is administratively down, the Unicast IPv6 addresses assigned to the interface are set to a pending state. Duplicate address detection is automatically restarted on an interface when the interface returns to being administratively up.

An interface returning to administratively up, restarts duplicate address detection for all of the Unicast IPv6 addresses on the interface. While duplicate address detection is performed on the link-local address of an interface, the state for the other IPv6 addresses is still set to `TENTATIVE`. When duplicate address detection is completed on the link-local address, duplicate address detection is performed on the remaining IPv6 addresses.

When duplicate address detection identifies a duplicate address, the state of the address 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 `SYSLOG` message is issued.

If the duplicate address is a global address of the interface, the address is not used and an error `SYSLOG` message is issued.

All configuration commands associated with the duplicate address remain as configured while the state of the address is set to `DUPLICATE`.

If the link-local address for an interface changes, duplicate address detection is performed on the new link-local address and all of the other IPv6 address associated with the interface are regenerated (duplicate address detection is performed only on the new link-local address).

Note. Since DAD is not supported on NBMA interfaces the command is allowed but does not impact on an IPv6 tunnel interface of the ISATAP type

it does not impact. The configuration is saved and will be impacted when the interface type is changed on another type on which DAD is supported (for example, to the IPv6 manual tunnel).

Example

The following example configures five consecutive neighbor solicitation messages to be sent on VLAN 1 while duplicate address detection is being performed on the tentative Unicast IPv6 address of the interface. The example also disables duplicate address detection processing on VLAN 2.

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd dad attempts 5
console(config-if)# exit
console(config)# interface vlan 2
console(config-if)# ipv6 nd dad attempts 0
console(config-if)# exit
```

ipv6 nd hop-limit

Use the `ipv6 nd hop-limit` command in Global Configuration mode to configure the maximum number of hops used in router advertisements.

To return the hop limit to its default value, use the `no` form of this command.

Syntax

```
ipv6 nd hop-limit value
```

```
no ipv6 nd hop-limit
```

Parameters

- *value*—Maximum number of hops. The acceptable range is from 1 to 255.

Default Configuration

The default value is defined by the `ipv6 hop-limit` command, or is set to 64 hops, if the command was not configured.

Command Mode

Interface Configuration mode

User Guidelines

Use this command if you want to change the default value. The default value is defined by the `ipv6 hop-limit` command.

Example

The following example configures a maximum number of 15 hops for router advertisements on VLAN 2:

```
console(config)# interface vlan 2
console(config-if)# ipv6 nd hop-limit 15
console(config-if)# exit
```

ipv6 nd managed-config-flag

Use the `ipv6 nd managed-config-flag` command in Interface Configuration mode to set the “managed address configuration flag” in IPv6 router advertisements.

To clear the flag from IPv6 router advertisements, use the `no` form of this command.

Syntax

```
ipv6 nd managed-config-flag
```

```
no ipv6 nd managed-config-flag
```

Parameters

N/A.

Default Configuration

The “managed address configuration flag” flag is not set in IPv6 router advertisements.

Command Mode

Interface Configuration mode

User Guidelines

Setting the Managed Address Configuration flag in IPv6 router advertisements indicates to attached hosts whether they should use stateful autoconfiguration to obtain addresses. If this flag is set, the attached hosts should use stateful autoconfiguration to obtain addresses, and if it is not set, the attached hosts should not use stateful autoconfiguration to obtain addresses.

Hosts may use stateful and stateless address autoconfiguration simultaneously.

Example

The following example configures the Managed Address Configuration flag in IPv6 router advertisements on VLAN 1:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd managed-config-flag
console(config-if)# exit
```

ipv6 nd ns-interval

Use the `ipv6 nd ns-interval` command in Interface Configuration mode to configure the interval between IPv6 neighbor solicitation retransmissions on an interface.

To restore the default interval, use the `no` form of this command.

Syntax

`ipv6 nd ns-interval` *milliseconds*

`no ipv6 nd ns-interval`

Parameters

- *milliseconds*—Interval between IPv6 neighbor solicit transmissions. The acceptable range is from 1000 to 3600000 milliseconds.

Default Configuration

0 seconds (unspecified) is advertised in router advertisements and the value 1000 milliseconds is used for the neighbor discovery activity of the router itself.

Command Mode

Interface Configuration mode

User Guidelines

This value will be included in all IPv6 router advertisements sent out this interface. Very short intervals are not recommended in normal IPv6 operation. When a non-default value is configured, the configured time is both advertised and used by the router itself.

Example

The following example configures an IPv6 neighbor solicit transmission interval of 9000 milliseconds for VLAN 1:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd ns-interval 9000
console(config-if)# exit
```

ipv6 nd other-config-flag

Use the `ipv6 nd other-config-flag` command in Interface Configuration mode to set the Other Stateful configuration flag in IPv6 router advertisements.

To clear the flag from IPv6 router advertisements, use the `no` form of this command.

Syntax

```
ipv6 nd other-config-flag
```

```
no ipv6 nd other-config-flag
```

Parameters

N/A.

Default Configuration

The Other Stateful configuration flag is not set in IPv6 router advertisements.

Command Mode

Interface Configuration mode

User Guidelines

The setting of the Other Stateful configuration flag in IPv6 router advertisements indicates to attached hosts how they can obtain autoconfiguration information other than addresses. If the flag is set, the attached hosts should use stateful autoconfiguration to obtain the other (nonaddress) information.

Note. If the Managed Address Configuration flag is set using the `ipv6 nd managed-config-flag` command, then an attached host can use stateful autoconfiguration to obtain the other (nonaddress) information regardless of the setting of the Other Stateful configuration flag.

Example

The following example configures the Other Stateful configuration flag in IPv6 router advertisements on VLAN 1:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd other-config-flag
console(config-if)# exit
```

ipv6 nd prefix

Use the `ipv6 nd prefix` command in Interface Configuration mode to configure which IPv6 prefixes are included in IPv6 Neighbor Discovery (ND) router advertisements.

To remove the prefixes, use the `no` form of this command.

Syntax

```
ipv6 nd prefix {ipv6-prefix/prefix-length | default} [no-advertise | {[valid-lifetime preferred-lifetime] [no-autoconfig] [off-link | no-onlink]}]
```

```
no ipv6 nd prefix [ipv6-prefix/prefix-length | default]
```

Parameters

- *ipv6-prefix*—IPv6 network number to include in router advertisements. This argument must be in the form documented in RFC4293, where the address is specified in hexadecimal using 16-bit values between colons.
- */prefix-length*—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 value.
- **default**—Default values used for automatic advertised prefixes configured as addresses on the interface using the `ipv6 address` command.
- **no-advertise**—Prefix is not advertised.
- *valid-lifetime*—Remaining length of time, in seconds, that this prefix will continue to be valid, i.e., time until invalidation. A value of 4,294,967,295 represents infinity. The address generated from an invalidated prefix should not appear as the destination or source address of a packet.
- *preferred-lifetime*—Remaining length of time, in seconds, that this prefix will continue to be preferred, i.e., time until deprecation. A value of 4,294,967,295 represents infinity. The address generated from a deprecated prefix should no longer be used as a source address in new communications, but packets received on such an interface are processed as expected. The *preferred-lifetime* must not be larger than the *valid-lifetime*.
- **no-autoconfig**—Indicates to hosts on the local link that the specified prefix cannot be used for IPv6 autoconfiguration. The prefix will be advertised with the A-bit clear.
- **off-link**—Configures the specified prefix as off-link. The prefix will be advertised with the L-bit clear. The prefix will not be inserted into the routing table as a connected prefix. If the prefix is already present in the routing table as a connected prefix (for example, because the prefix was also configured using the `ipv6 address` command), then it will be removed.
- **no-onlink**—Configures the specified prefix as not on-link. The prefix will be advertised with the L-bit clear.

Default Configuration

All prefixes configured on interfaces that originate IPv6 router advertisements are advertised with a valid lifetime of 2,592,000 seconds (30 days) and a preferred lifetime of 604,800 seconds (7 days).

Note that by default:

- All prefixes are inserted in the routing table as connected prefixes.
- All prefixes are advertised as on-link (for example, the L-bit is set in the advertisement)
- All prefixes are advertised as an auto-configuration prefix (for example, the A-bit is set in the advertisement)

Command Mode

Interface Configuration mode

User Guidelines

This command enables control over the individual parameters per prefix, including whether the prefix should be advertised.

Use the **ipv6 nd prefix** *ipv6-prefix/prefix-length* command to add the prefix to the Prefix table.

Use the **no ipv6 nd prefix** *ipv6-prefix/prefix-length* command to remove the prefix from the Prefix table.

Use the **no ipv6 nd prefix** command without the *ipv6-prefix/prefix-length* argument to remove all prefixes from the Prefix Table.

Note that the **no ipv6 nd prefix** command does not return the default values to the original default values.

The switch supports the following advertisement algorithm:

- Advertise all prefixes that are configured as addresses on the interface using the parameters defined by the **ipv6 nd prefix default** command (or the default value if the command has not been configured) except prefixes that are placed in the Prefix table (changed (configured) by the **ipv6 nd prefix** command).
- Advertise all prefixes configured by the **ipv6 nd prefix** command without the **no-advertise** keyword.

Default Keyword

The **default** keyword can be used to set default values for automatic advertised prefixes configured as addresses on the interface using the **ipv6 address** command.

Note. These default values are not used as the default values in the **ipv6 nd prefix** command.

Use the **no ipv6 nd prefix default** command to return the default values to the original default values.

On-Link

When on-link is “on” (by default), the specified prefix is assigned to the link. Nodes sending traffic to such addresses that contain the specified prefix consider the destination to be locally reachable on the link. An on-link prefix is inserted into the routing table as a Connected prefix.

Auto-configuration

When auto-configuration is on (by default), it indicates to hosts on the local link that the specified prefix can be used for IPv6 auto-configuration.

The configuration options affect the L-bit and A-bit settings associated with the prefix in the IPv6 ND Router Advertisement, and presence of the prefix in the routing table, as follows:

- **Default** L=1 A=1, In the Routing Table
- **no-onlink** L=0 A=1, In the Routing Table
- **no-autoconfig** L=1 A=0, In the Routing Table
- **no-onlink no-autoconfig** L=0 A=0, In the Routing Table
- **off-link** L=0 A=1, Not in the Routing Table
- **off-link no-autoconfig** L=0 A=0, Not in the Routing Table

Example

Example 1. The following example includes the IPv6 prefix 2001:0DB8::/35 in router advertisements sent out VLAN 1 with a valid lifetime of 1000 seconds and a preferred lifetime of 900 seconds. The prefix is inserted in the Routing table:

```
console(config)# interface vlan 1
```

```
console(config-if)# ipv6 nd prefix 2001:0DB8::/35 1000 900  
console(config-if)# exit
```

Example 2. The following example advertises the prefix with the L-bit clear:

```
console(config)# interface vlan 1  
console(config-if)# ipv6 address 2001::1/64  
console(config-if)# ipv6 nd prefix 2001::/64 3600 3600 no-  
onlink  
console(config-if)# exit
```

ipv6 nd ra interval

Use the **ipv6 nd ra interval** command in Interface Configuration mode to configure the interval between IPv6 router advertisement (RA) transmissions on an interface.

To restore the default interval, use the **no** form of this command.

Syntax

```
ipv6 nd ra interval maximum-secs [minimum-secs]
```

```
no ipv6 nd ra interval
```

Parameters

- *maximum-secs*—Maximum interval between IPv6 RA transmissions in seconds. The range is from 4 to 1800.
- *minimum-secs*—Minimum interval between IPv6 RA transmissions in seconds. The range is from 3 to 1350.

Default Configuration

maximum-secs is 600 seconds.

minimum-secs is $0.33 * \text{maximum-secs}$, if the value ≥ 3 seconds and is 3 seconds, if the value < 3 seconds.

Command Mode

Interface Configuration mode

User Guidelines

The interval between transmissions should be less than or equal to the IPv6 router advertisement lifetime if you configure the route as a default router by using this command. To prevent synchronization with other IPv6 nodes, the actual interval used is randomly selected from a value between the minimum and maximum values.

The minimum RA interval may never be more than 75% of the maximum RA interval and never less than 3 seconds.

Example

Example 1. The following example configures an IPv6 router advertisement interval of 201 seconds for VLAN 1:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd ra interval 201
console(config-if)# exit
```

Example 2. The following examples shows a maximum RA interval of 200 seconds and a minimum RA interval of 50 seconds:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd ra interval 200 50
console(config-if)# exit
```

ipv6 nd ra lifetime

Use the `ipv6 nd ra lifetime` command in Interface Configuration mode to configure the Router Lifetime value in IPv6 router advertisements on an interface.

To restore the default lifetime, use the `no` form of this command.

Syntax

`ipv6 nd ra lifetime seconds`

`no ipv6 nd ra lifetime`

Parameters

- *seconds*—Remaining length of time, in seconds, that this router will continue to be useful as a default router (Router Lifetime value). A value of zero indicates that it is no longer useful as a default router. The acceptable range is 0 or from <Maximum RA Interval> to 9000 seconds.

Default Configuration

The default lifetime value is 3* <Maximum RA Interval> seconds.

Command Mode

Interface Configuration mode

User Guidelines

The Router Lifetime value is included in all IPv6 router advertisements sent out the interface. The value indicates the usefulness of the router as a default router on this interface. Setting the value to 0 indicates that the router should not be considered a default router on this interface. The Router Lifetime value can be set to a non-zero value to indicate that it should be considered a default router on this interface. The non-zero value for the Router Lifetime value should not be less than the router advertisement interval.

Example

The following example configures an IPv6 router advertisement lifetime of 1801 seconds for VLAN 1:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd ra lifetime 1801
console(config-if)# exit
```

ipv6 nd ra suppress

Use the `ipv6 nd ra suppress` command in Interface Configuration mode to suppress IPv6 router advertisement transmissions on an interface. To re-enable the sending of IPv6 router advertisement transmissions on an interface, use the `no` form of this command.

Syntax

`ipv6 nd ra suppress`

`no ipv6 nd ra suppress`

Parameters

N/A.

Default Configuration

LAN interface - IPv6 router advertisements are automatically sent.

Point-to-Point interface - IPv6 router advertisements are suppressed.

NBMA interface - IPv6 router advertisements are suppressed.

Command Mode

Interface Configuration mode

User Guidelines

Use the `no ipv6 nd ra suppress` command to enable the sending of IPv6 router advertisement transmissions on a Point-to-Point interface (for example, manual tunnel).

Use the `no ipv6 nd ra suppress` command to enable the sending of IPv6 router advertisement transmissions on a NBMA interface (for example, ISATAP tunnel).

Example

Example 1. The following example suppresses IPv6 router advertisements on vlan 1:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd ra suppress
console(config-if)# exit
```

Example 2. The following example enables the sending of IPv6 router advertisements on tunnel 1:

```
console(config)# interface tunnel 1
```

```
console(config-if)# no ipv6 nd ra suppress
console(config-if)# exit
```

ipv6 nd reachable-time

Use the `ipv6 nd reachable-time` command in Interface Configuration mode to configure the amount of time that a remote IPv6 node is considered reachable after some reachability confirmation event has occurred.

To restore the default time, use the `no` form of this command.

Syntax

```
ipv6 nd reachable-time milliseconds
```

```
no ipv6 nd reachable-time
```

Parameters

- *milliseconds*—Amount of time that a remote IPv6 node is considered reachable (in milliseconds). The acceptable range is from 0 to 3600000 milliseconds.

Default Configuration

0 milliseconds (unspecified) is advertised in router advertisements and the value 30000 (30 seconds) is used for the neighbor discovery activity of the router itself.

Command Mode

Interface Configuration mode

User Guidelines

The configured time enables the router to detect unavailable neighbors. Shorter configured times enable the router to detect unavailable neighbors more quickly; however, shorter times consume more IPv6 network bandwidth and processing resources in all IPv6 network devices. Very short configured times are not recommended in normal IPv6 operation.

The configured time is included in all router advertisements sent out of an interface so that nodes on the same link use the same time value. A value of 0 means indicates that the configured time is unspecified by this router.

Example

The following example configures an IPv6 reachable time of 1,700,000 milliseconds for VLAN 1:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd reachable-time 1700000
console(config-if)# exit
```

ipv6 nd router-preference

Use the `ipv6 nd router-preference` command in Interface Configuration mode to configure a default router preference (DRP) for the router on a specific interface.

To return to the default DRP, use the `no` form of this command.

Syntax

```
ipv6 nd router-preference {high | medium | low}
```

```
no ipv6 nd router-preference
```

Parameters

- **high**—Preference for the router specified on an interface is high.
- **medium**—Preference for the router specified on an interface is medium.
- **low**—Preference for the router specified on an interface is low.

Default Configuration

Router advertisements (RAs) are sent with the medium preference.

Command Mode

Interface Configuration mode

User Guidelines

RA messages are sent with the DRP configured by the this command. If no DRP is configured, RAs are sent with a medium preference.

A DRP is useful when, for example, two routers on a link may provide equivalent, but not equal-cost, routing, and policy may dictate that hosts should prefer one of the routers.

Example

The following example configures a DRP of high for the router on VLAN 1:

```
console(config)# interface vlan 1
console(config-if)# ipv6 nd router-preference high
console(config-if)# exit
```

ipv6 neighbor

Use the **ipv6 neighbor** command in Global Configuration mode to configure a static entry in the IPv6 neighbor discovery cache. To remove a static IPv6 entry from the IPv6 neighbor discovery cache, use the **no** form of this command.

Syntax

```
ipv6 neighbor ipv6-address interface-id mac-address
no ipv6 neighbor [ipv6-address] interface-id
```

Parameters

- *ipv6-address*—Specified IPv6 address. This argument must be in the form documented in RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.
- *interface-id*—Specified interface identifier.
- *mac-address*—Interface MAC address.

Default Configuration

Static entries are not configured in the IPv6 neighbor discovery cache.

Command Mode

Global Configuration mode

User Guidelines

This command is similar to the `arp` (global) command.

Use the `ipv6 neighbor` command to add a static entry in the IPv6 neighbor discovery cache.

If the specified IPv6 address is a global IPv6 address it must belong to one of static on-link prefixes defined in the interface. When a static on-link prefix is deleted all static entries in the IPv6 neighbor discovery cache corresponding the prefix is deleted to.

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.

Static entries in the IPv6 neighbor discovery cache are not modified by the neighbor discovery process.

Use the `no ipv6 neighbor ipv6-address interface-id` command to remove the one given static entry on the given interface. The command does not remove the entry from the cache, if it is a dynamic entry, learned from the IPv6 neighbor discovery process.

Use the `no ipv6 neighbor interface-id` command to delete the all static entries on the given interface.

Use the `no ipv6 neighbor` command to remove the all static entries on all interfaces.

Use the `show ipv6 neighbors` command to view static entries in the IPv6 neighbor discovery cache. A static entry in the IPv6 neighbor discovery cache can have one of the following states:

- `NCMP` (Incomplete)—The interface for this entry is down.
- `REACH` (Reachable)—The interface for this entry is up.

Note. Reachability detection is not applied to static entries in the IPv6 neighbor discovery cache; therefore, the descriptions for the `INCMP` and `REACH` states are different for dynamic and static cache entries.

Example

Example 1. The following example configures a static entry in the IPv6 neighbor discovery cache for a neighbor with the IPv6 address 2001:0DB8::45A and link-layer address 0002.7D1A.9472 on VLAN 1:

```
console(config)# ipv6 neighbor 2001:0DB8::45A vlan1  
0002.7D1A.9472
```

Example 2. The following example deletes the static entry in the IPv6 neighbor discovery cache for a neighbor with the IPv6 address 2001:0DB8::45A and link-layer address 0002.7D1A.9472 on VLAN 1:

```
console(config)# no ipv6 neighbor 2001:0DB8::45A vlan1
```

Example 3. The following example deletes all static entries in the IPv6 neighbor discovery cache on VLAN 1:

```
console(config)# no ipv6 neighbor vlan1
```

Example 4. The following example deletes all static entries in the IPv6 neighbor discovery cache on all interfaces:

```
console(config)# no ipv6 neighbor
```

ipv6 redirects

Use the **ipv6 redirects** command in Interface Configuration mode to enable the sending of ICMP IPv6 redirect messages to re-send a packet through the same interface on which the packet was received.

To disable the sending of redirect messages, use the **no** form of this command.

Syntax

ipv6 redirects

no ipv6 redirects

Parameters

N/A.

Default Configuration

The sending of ICMP IPv6 redirect messages is enabled.

Command Mode

Interface Configuration mode

Example

The following example disables the sending of ICMP IPv6 redirect messages on VLAN 100 and re-enables the messages on VLAN 2:

```
console(config)# interface vlan 100
console(config-if)# no ipv6 redirects
console(config-if)# exit
console(config)# interface vlan 2
console(config-if)# ipv6 redirects
console(config-if)# exit
```

ipv6 route

Use the **ipv6 route** command in Global Configuration mode to establish static IPv6 routes.

To remove a previously configured static route, use the **no** form of this command.

Syntax

```
ipv6 route ipv6-prefix/prefix-length next--ipv6-address [metric]
```

```
no ipv6 route ipv6-prefix/prefix-length next--ipv6-address
```

Parameters

- *ipv6-prefix*—IPv6 network that is the destination of the static route. Can also be a host name when static host routes are configured.

- */prefix-length*—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 value.
- *next-ipv6-address*—IPv6 address of the next hop that can be used to reach the specified network.
 - If the *next-ipv6-address* argument is a link local address it must be defined in the zone format: IPv6 Zone Format > ::= IPv6-Link-Local-Address%Interface-ID
 - The *interface-id* argument must be coded without spaces.
- *metric*—Static route metric. Acceptable values are from 1 to 65535. The default value is 1.

Default Configuration

Static entries are not configured in the IPv6 neighbor discovery cache.

Command Mode

Global Configuration mode

User Guidelines

If the next IPv6 address is a global IPv6 address, it should belong to a static on-link prefix. When an on-link prefix is removed or is changed to non on-link prefix, the static routes with next hop belonging to the prefix are removed from the configuration.

Examples

Example 1. The following example defines a static route with a global next hop:

```
console(config)# ipv6 route 2001::/64 5::5 10
```

Example 2. The following example defines a static route with a link-local next hop:

```
console(config)# ipv6 route 2001:DB8:2222::/48
FE80::260:3EFF:FE11:6770%vlan1 12
```

ipv6 unicast-routing

Use the **ipv6 unicast-routing** command in Global Configuration mode to enable the forwarding of IPv6 Unicast datagrams.

To disable the forwarding of IPv6 Unicast datagrams, use the **no** form of this command.

Syntax

ipv6 unicast-routing

no ipv6 unicast-routing

Parameters

N/A.

Default Configuration

IPv6 Unicast routing is disabled.

Command Mode

Global Configuration mode

Example

The following example enables the forwarding of IPv6 Unicast datagrams:

```
console(config)# ipv6 unicast-routing
```

ipv6 unreachable

Use the **ipv6 unreachable** command in Interface Configuration mode to enable the generation of Internet Control Message Protocol for IPv6 (ICMPv6) unreachable messages for any packets arriving on a specified interface.

To prevent the generation of unreachable messages, use the **no** form of this command.

Syntax

ipv6 unreachable

no ipv6 unreachable

Parameters

N/A.

Default Configuration

The sending of ICMP IPv6 unreachable messages is enabled.

Command Mode

Interface Configuration mode

User Guidelines

If the switch receives a Unicast packet destined for itself that uses a protocol it does not recognize, it sends an ICMPv6 unreachable message to the source.

If the switch receives a datagram that it cannot deliver to its ultimate destination because it knows of no route to the destination address, it replies to the originator of that datagram with an ICMP host unreachable message.

Example

The following example disables the generation of ICMPv6 unreachable messages, as appropriate, on an interface:

```
console(config)# interface vlan 100
console(config-if)# no ipv6 unreachable
console(config-if)# exit
```

show ipv6 interface

Use the `show ipv6 interface` command in user EXEC or privileged EXEC mode to display the usability status of interfaces configured for IPv6.

Syntax

show ipv6 interface [brief] | [[*interface-id*] [prefix]]

Parameters

- **brief**—Displays a brief summary of IPv6 status and configuration for each interface where IPv6 is defined.
- *interface-id*—Interface identifier about which to display information.
- **prefix**—Prefix generated from a local IPv6 prefix pool.

Default Configuration

Option **brief** - all IPv6 interfaces are displayed.

Command Mode

User EXEC mode

Privileged EXEC mode

User Guidelines

Use this command to validate the IPv6 status of an interface and its configured addresses. This command also displays the parameters that IPv6 uses for operation on this interface and any configured features.

If the interface's hardware is usable, the interface is marked up.

If you specify an optional interface identifier, the command displays information only about that specific interface. For a specific interface, you can enter the **prefix** keyword to see the IPv6 neighbor discovery (ND) prefixes that are configured on the interface.

Example

Example 1. The `show ipv6 interface` command displays information about the specified interface:

```
console# show ipv6 interface vlan 1
VLAN 1 is up/up
IPv6 is enabled, link-local address is FE80::0DB8:12AB:FA01
IPv6 Forwarding is enabled
Global unicast address(es):
IPv6 Global Address                               Type
2000:0DB8::2/64 (ANY)                             Manual
```

```
2000:0DB8::2/64 Manual
2000:1DB8::2011/64 Manual
Joined group address(es):
FF02::1
FF02::2
FF02::1:FF11:6770
MTU is 1500 bytes
ICMP error messages limited interval is 100ms; Bucket size is 10
tokens
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router maximum advertisement interval is 600 seconds
ND router minimum advertisement interval is 198 seconds (DEFAULT)
ND router advertisements live for 1800 seconds
ND advertised default router preference is Medium
Stateless autoconfiguration is enabled.
MLD Version is 2
```

Field Descriptions:

- **vlan 1 is up/up**—Indicates the interface status: administrative/operational.
- **IPv6 is enabled, stalled, disabled (stalled and disabled are not shown in sample output)**—Indicates that IPv6 is enabled, stalled, or disabled on the interface. If IPv6 is enabled, the interface is marked Enabled. If duplicate address detection processing identified the link-local address of the interface as being a duplicate address, the processing of IPv6 packets is disabled on the interface and the interface is marked Stalled. If IPv6 is not enabled, the interface is marked Disabled.
- **link-local address**—Displays the link-local address assigned to the interface.
- **Global unicast address(es):**—Displays the global Unicast addresses assigned to the interface. The type is **manual** or **autoconfig**.

- **Joined group address(es):**—Indicates the Multicast groups to which this interface belongs.
- **ICMP error messages**—Specifies the minimum interval (in milliseconds) between error messages sent on this interface.
- **ICMP redirects**—State of ICMP IPv6 redirect messages on the interface (the sending of the messages is enabled or disabled).
- **ND DAD**—The state of duplicate address detection on the interface (enabled or disabled).
- **number of DAD attempts:**—Number of consecutive neighbor solicitation messages that are sent on the interface while duplicate address detection is performed.
- **ND reachable time**—Displays the neighbor discovery reachable time (in milliseconds) assigned to this interface.
- **ND advertised reachable time**—Displays the neighbor discovery reachable time (in milliseconds) advertised on this interface.
- **ND advertised retransmit interval**—Displays the neighbor discovery retransmit interval (in milliseconds) advertised on this interface.
- **ND router advertisements**—Specifies the interval (in seconds) for neighbor discovery router advertisements sent on this interface and the amount of time before the advertisements expire.
- **ND advertised default router preference is Medium**—DRP for the router on a specific interface.
- **MLD Version**—Version of MLD

Example 2. The `show ipv6 interface` command displays information about the specified manual IPv6 tunnel:

```

console# show ipv6 interface tunnel 2
Tunnel 2 is up/up
IPv6 is enabled, link-local address is FE80::0DB8:12AB:FA01
IPv6 Forwarding is enabled
Global unicast address(es):
IPv6 Global Address                               Type

```

```

2000:0DB8::2/64 (ANY)                               Manual
2000:0DB8::2/64                                     Manual
2000:1DB8::2011/64                                  Manual
Joined group address(es):
FF02::1
FF02::2
FF02::1:FF11:6770
MTU is 1500 bytes
ICMP error messages limited interval is 100ms; Bucket size is 10
tokens
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
ND advertised default router preference is Medium
Hosts use stateless autoconfig for addresses.
MLD Version is 2
Tunnel mode is manual
Tunnel Local IPv4 address : 10.10.10.1(auto)
Tunnel Remote Ipv4 address : 10.1.1.1

```

Field Descriptions:

- **vlan 1 is up/up**—Indicates the interface status: administrative/operational.
- **IPv6 is enabled, stalled, disabled (stalled and disabled are not shown in sample output)**—Indicates that IPv6 is enabled, stalled, or disabled on the interface. If IPv6 is enabled, the interface is marked “enabled.” If duplicate address detection processing identified the link-local address of the interface as being a duplicate address, the processing of IPv6 packets is disabled on the interface and the interface is marked “stalled.” If IPv6 is not enabled, the interface is marked “disabled.”

- **link-local address**—Displays the link-local address assigned to the interface.
- **Global Unicast address(es)**:—Displays the global Unicast addresses assigned to the interface. The type is **manual** or **autoconfig**.
- **Joined group address(es)**:—Indicates the Multicast groups to which this interface belongs.
- **ICMP error messages**—Specifies the minimum interval (in milliseconds) between error messages sent on this interface.
- **ICMP redirects**—The state of Internet Control Message Protocol (ICMP) IPv6 redirect messages on the interface (the sending of the messages is enabled or disabled).
- **ND DAD**—The state of duplicate address detection on the interface (enabled or disabled).
- **number of DAD attempts**:—Number of consecutive neighbor solicitation messages that are sent on the interface while duplicate address detection is performed.
- **ND reachable time**—Displays the neighbor discovery reachable time (in milliseconds) assigned to this interface.
- **ND advertised reachable time**—Displays the neighbor discovery reachable time (in milliseconds) advertised on this interface.
- **ND advertised retransmit interval**—Displays the neighbor discovery retransmit interval (in milliseconds) advertised on this interface.
- **ND router advertisements**—Specifies the interval (in seconds) for neighbor discovery router advertisements sent on this interface and the amount of time before the advertisements expire.
- **ND advertised default router preference is Medium**—The DRP for the router on a specific interface.
- **MLD Version**—The version of MLD
- **Tunnel mode**—Specifies the tunnel mode: **manual**, **6to4**, **auto-tunnel** or **isatap**
- **Tunnel Local IPv4 address**—Specifies the tunnel local IPv4 address and have one of the following formats:
 - ipv4-address

- *ipv4-address* (auto)
 - *ipv4-address* (*interface-id*)
 - **Tunnel Remote IPv4 address**—Specifies the tunnel remote IPv4 address
-

Example 3. The `show ipv6 interface` command displays information about the specified ISATAP tunnel:

```
console# show ipv6 interface tunnel 1
Tunnel 1 is up/up
IPv6 is enabled, link-local address is FE80::0DB8:12AB:FA01
ICMP redirects are disabled
Global unicast address(es):
Ipv6 Global Address                                Type
2000:0DB8::2/64 (ANY)                              Manual
2000:0DB8::2/64                                    Manual
2000:1DB8::2011/64                                 Manual
Joined group address(es):
FF02::1
FF02::2
FF02::1:FF11:6770
    is 1500 bytes
ICMP error messages limited interval is 100ms; Bucket size is 10
tokens
ICMP redirects are enabled
ND DAD is disabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
ND advertised default router preference is Medium
Hosts use stateless autoconfig for addresses.
MLD Version is 2
```

Tunnel mode is ISATAP

Tunnel Local IPv4 address : 10.10.10.1(VLAN 1)

ISATAP Router DNS name is isatap

Field Descriptions:

- **ND DAD**—The state of duplicate address detection on the interface (enabled or disabled). **Note.** The state of duplicate address detection on an IPv6 tunnel interface of ISATAP type always is displayed as disabled regardless of a value of the **number of DAD attempts** parameter because DAD is not supported on NBMA interfaces. The switch will enable DAD automatically when the user change the type of the tunnel to manual if a the parameter value bigger than 0.
- **number of DAD attempts**:—Number of consecutive neighbor solicitation messages that are sent on the interface while duplicate address detection is performed.
- **vlan 1 is up/up**—Indicates the interface status: administrative/operational.
- **IPv6 is enabled, stalled, disabled (stalled and disabled are not shown in sample output)**—Indicates that IPv6 is enabled, stalled, or disabled on the interface. If IPv6 is enabled, the interface is marked “enabled.” If duplicate address detection processing identified the link-local address of the interface as being a duplicate address, the processing of IPv6 packets is disabled on the interface and the interface is marked “stalled.” If IPv6 is not enabled, the interface is marked “disabled.”
- **link-local address**—Displays the link-local address assigned to the interface.
- **Global Unicast address(es)**:—Displays the global Unicast addresses assigned to the interface. The type is **manual** or **autoconfig**.
- **Joined group address(es)**:—Indicates the Multicast groups to which this interface belongs.
- **ICMP error messages**—Specifies the minimum interval (in milliseconds) between error messages sent on this interface.
- **ICMP redirects**—The state of Internet Control Message Protocol (ICMP) IPv6 redirect messages on the interface (the sending of the messages is enabled or disabled).

- **number of DAD attempts:**—Number of consecutive neighbor solicitation messages that are sent on the interface while duplicate address detection is performed.
- **ND reachable time**—Displays the neighbor discovery reachable time (in milliseconds) assigned to this interface.
- **ND advertised reachable time**—Displays the neighbor discovery reachable time (in milliseconds) advertised on this interface.
- **ND advertised retransmit interval**—Displays the neighbor discovery retransmit interval (in milliseconds) advertised on this interface.
- **ND router advertisements**—Specifies the interval (in seconds) for neighbor discovery router advertisements sent on this interface and the amount of time before the advertisements expire.
- **ND advertised default router preference is Medium**—The DRP for the router on a specific interface.
- **MLD Version**—The version of MLD
- **Tunnel mode**—Specifies the tunnel mode: **manual**, **6to4**, **auto-tunnel** or **isatap**
- **Tunnel Local IPv4 address**—Specifies the tunnel local IPv4 address and have one of the following formats:
 - `ipv4-address`
 - `ipv4-address (auto)`
 - `ipv4-address (interface-id)`
- **Tunnel Remote IPv4 address**—Specifies the tunnel remote IPv4 address
- **ISATAP Router DNS name is**—The DNS name of the ISATAP Router

Example 4. The following command with the **brief** keyword displays information about all interfaces that IPv6 is defined on:

```

console# show ipv6 interface brief
Interface Interface IPv6      Link Local          Number of
           State   State   IPv6 Address        Global Addresses
-----
po1         down/down enabled  FE80::0DB8:12AB:FA01  2

```

tunnel 1	up/up	enabled	FE80::0DB8:12AB:FA01	1
vlan 1	up/up	enabled	FE80::0DB8:12AB:FA01	3
vlan 1000	up/up	stalled	FE80::0DB8:12AB:FA01	2

Example 5. This sample output shows the characteristics of VLAN 1 that has generated a prefix from a local IPv6 prefix pool:

```

console# configure terminal
console(config)# interface vlan1
console(config-if)# ipv6 address 2001:0DB8:1::1/64
console(config-if)# ipv6 address 2001:0DB8:2::1/64
console(config-if)# ipv6 address 2001:0DB8:3::1/64
console(config-if)# ipv6 nd prefix 2001:0DB8:1::/64 no-advertise
console(config-if)# ipv6 nd prefix 2001:0DB8:3::/64 2912000
564900 off-link
console(config-if)# ipv6 nd prefix 2001:0DB8:4::/64
console(config-if)# ipv6 nd prefix 2001:0DB8:5::/64 2912000
564900 off-link
console(config-if)# exit
console(config)# exit
console# show ipv6 interface vlan 1 prefix
IPv6 Prefix Advertisements VLAN 1
Codes: A - Address, P - Prefix is advertised, R is in Routing Table
Code Prefix                Flags Valid Lifetime Preferred Lifetime
-----
      default                LA      2592000      604800
AR  2001:0DB8:1::/64        LA      infinite      infinite
APR 2001:0DB8:2::/64        LA      infinite      infinite
AP  2001:0DB8:3::/64        A       infinite      infinite
PR  2001:0DB8:4::/64        LA      2592000      604800
P   2001:0DB8:5::/64        A       2912000      564900

```

show ipv6 link-local default zone

Use the `show ipv6 link-local default zone` command in user EXEC or privileged EXEC mode to display the IPv6 link local default zone.

Syntax

`show ipv6 link-local default zone`

Command Mode

User EXEC mode

Privileged EXEC mode

Example

Example 1. The following example displays the default zone when it is defined:

```
console# show ipv6 link-local default zone
Link Local Default Zone is VLAN 1
```

Example 2. The following example displays the default zone when it is not defined:

```
console# show ipv6 link-local default zone
Link Local Default Zone is not defined
```

show ipv6 nd prefix

Use the `show ipv6 nd prefix` command in user EXEC or privileged EXEC mode to display IPv6 prefixes included in IPv6 Neighbor Discovery (ND) router advertisements.

Syntax

`show ipv6 nd prefix [interface-id]`

Parameters

- *interface-id*—Specified interface identifier on which prefixes are advertised.

Default Configuration

No prefixes are displayed.

Command Mode

User EXEC mode

Privileged EXEC mode

User Guidelines

Use the **how ipv6 nd prefix** command with the *interface-id* argument to display prefixes advertised on a single interface.

Example

The following example displays IPv6 prefixes:

```
console# show ipv6 nd prefix vlan 100
vlan 100
  default
    valid-lifetime 2,592,000 secs
    preferred-lifetime 604,800 secs
    on-link
    auto-config
  prefix 2001::1/64
    valid-lifetime 3,600 secs
    preferred-lifetime 2,700 secs
  prefix 2001:2:12/64
    no advertise
  prefix 2002::1/64
    valid-lifetime 3,600 secs
    preferred-lifetime 2,700 secs
    on-link
```

```
prefix 2011::1/64
  valid-lifetime 3,600 secs
  preferred-lifetime 2,700 secs
  off-link
  auto-config
```

show ipv6 neighbors

Use the `show ipv6 neighbors` command in User EXEC or Privileged EXEC mode to display IPv6 neighbor discovery (ND) cache information.

Syntax

```
show ipv6 neighbors [interface-id | ipv6-address | ipv6-hostname]
```

Parameters

- *interface-id*—Specifies the identifier of the interface from which IPv6 neighbor information is to be displayed.
- *ipv6-address*—Specifies the IPv6 address of the neighbor. This argument must be in the form documented in RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.
- *ipv6-hostname*—Specifies the IPv6 host name of the remote networking device.

Default Configuration

All IPv6 ND cache entries are listed.

Command Mode

User EXEC mode

Privileged EXEC mode

User Guidelines

When the *interface-id* argument is not specified, cache information for all IPv6 neighbors is displayed. Specifying the *interface-id* argument displays only cache information about the specified interface.

Example

Example 1. The following is sample output from the `show ipv6 neighbors` command when entered with an interface-id:

```
console# show ipv6 neighbors vlan 1
IPv6 Address          Age Link-layer Addr    State  Interface
Router
2000:0:0:4::2        0   0003.a0d6.141e    REACH  VLAN1
Yes
3001:1::45a          -   0002.7d1a.9472    REACH  VLAN1
-
FE80::203:A0FF:FED6:141E  0   0003.a0d6.141e    REACH  VLAN1
No
```

Example 2. The following is sample output from the `show ipv6 neighbors` command when entered with an IPv6 address:

```
console# show ipv6 neighbors 2000:0:0:4::2
IPv6 Address          Age Link-layer Addr    State  Interface
Router
2000:0:0:4::2        0   0003.a0d6.141e    REACH  VLAN1
Yes
```

Field Descriptions:

- **Total number of entries**—Number of entries (peers) in the cache.
- **IPv6 Address**—IPv6 address of neighbor or interface.
- **Age**—Time (in minutes) since the address was confirmed to be reachable. A hyphen (-) indicates a static entry.
- **Link-layer Addr**—MAC address. If the address is unknown, a hyphen (-) is displayed.
- **Interface**—Interface which the neighbor is connected to.
- **Router**—Specifies if the neighbor is a Router. A hyphen (-) is displayed for static entries.

show ipv6 route

Use the **show ipv6 route** command in user EXEC or privileged EXEC mode to display the current contents of the IPv6 routing table.

Syntax

```
show ipv6 route [ipv6-address | ipv6-prefix/prefix-length | protocol |  
interface interface-id]
```

Parameters

- *ipv6-address*—Displays routing information for a specific IPv6 address. This argument must be in the form documented in RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.
- *ipv6-prefix*—Displays routing information for a specific IPv6 network. This argument must be in the form documented in RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.
- */prefix-length*—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 value.
- *protocol*—Displays routes for the specified routing protocol using any of these keywords: **bgp**, **isis**, **ospf**, or **rip**; or displays routes for the specified type of route using any of these keywords: **connected**, **static**, **nd**, or **icmp**.
- *interface interface-id*—Identifier of an interface.

Default Configuration

All IPv6 routing information for all active routing tables is displayed.

Command Mode

User EXEC mode

Privileged EXEC mode

User Guidelines

This command provides output similar to the **show ip route** command, except that the information is IPv6-specific.

When the *ipv6-address* or *ipv6-prefix/prefix-length* argument is specified, a longest match lookup is performed from the routing table and only route information for that address or network is displayed. When the **icmp**, **nd**, **connected**, **local**, or **static** keywords are specified, only that type of route is displayed. When the *interface-id* argument are specified, only the specified interface-specific routes are displayed.

Example

Example 1. The following is sample output from the **show ipv6 route** command when IPv6 Routing is not enabled and the command is entered without an IPv6 address or prefix specified:

```
console# show ipv6 route
Codes: > - Best
        S - Static, I - ICMP Redirect, ND - Router Advertisement
[d/m]: d - route's distance, m - route's metric
IPv6 Routing Table - 6 entries
S> ::/0 [1/1]
    via fe80::77 VLAN 1
ND> ::/0 [11/0]
    via fe80::200:cff:fe4a:dfa8 VLAN 1 Lifetime 1784 sec
ND> 2001::/64 [0/0]
    via :: VLAN 100
ND> 2002:1:1:1::/64 [0/0]
    via :: VLAN 100
ND> 3001::/64 [0/0]
    via :: VLAN 101
ND> 4004::/64 [0/0]
    via :: VLAN 110
```

Example 2. The following is sample output from the **show ipv6 route** command when IPv6 Routing is enabled and the command is entered without an IPv6 address or prefix specified and IPv6 Routing is enabled:

```
console# show ipv6 route
```

Codes: > - Best

S - Static, C - Connected,

L - Local(on-link prefixes defined by the **ipv6 nd prefix** command with **on-link** keyword,

[d/m]: d - route's distance, m - route's metric

IPv6 Routing Table - 3 entries

```
S> 3000::/64 [1/1]
    via FE80::A8BB:CCFF:FE02:8B00   VLAN 100
C> 4001::/64 [0/0]
    via ::   VLAN 100
L> 4002::/64 [0/0]
    via ::   VLAN 100 Lifetime 9000 sec
```

show ipv6 route summary

Use the **show ipv6 route summary** command in User EXEC or Privileged EXEC mode to display the current contents of the IPv6 routing table in summary format.

Syntax

show ipv6 route summary

Parameters

N/A.

Command Mode

User EXEC mode

Privileged EXEC mode

Example

The following is sample output from the **show ipv6 route summary** command:

```
console# show ipv6 route summary
```

```
IPv6 Routing Table Summary - 97 entries
37 local, 35 connected, 25 static
Number of prefixes:
/16: 1, /28: 10, /32: 5, /35: 25, /40: 1, /64: 9
/96: 5, /112: 1, /127: 4, /128: 36
```

show ipv6 static

Use the `show ipv6 static` command in user EXEC or privileged EXEC mode to display the current static routes of the IPv6 routing table.

Syntax

```
show ipv6 static [ipv6-address | ipv6-prefix/prefix-length] [interface
interface-id] [detail]
```

Parameters

- *ipv6-address*—Provides routing information for a specific IPv6 address. This argument must be in the form documented in RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.
- *ipv6-prefix*—Provides routing information for a specific IPv6 network. This argument must be in the form documented in RFC4293 where the address is specified in hexadecimal using 16-bit values between colons.
- */prefix-length*—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 value.
- *interface interface-id*—Identifier of an interface.
- *detail*—Specifies for invalid routes, the reason why the route is not valid.

Default Configuration

All IPv6 static routing information for all active routing tables is displayed.

Command Mode

User EXEC mode

Privileged EXEC mode

User Guidelines

When the *ipv6-address* or *ipv6-prefix/prefix-length* argument is specified, a longest match lookup is performed from the routing table and only route information for that address or network is displayed. Only the information matching the criteria specified in the command syntax is displayed. For example, when the *interface-id* argument is specified, only the specified interface-specific routes are displayed.

When the **detail** keyword is specified, the reason why the route is not valid is displayed for invalid direct or fully specified routes.

Example

Example 1. The following is sample output from the `show ipv6 static` command without specified options:

```
console# show ipv6 static
IPv6 Static routes   Code: * - installed in Routing Information
Base (RIB)
IPv6 Static routes distance is 1
* 3000::/16, interface VLAN1, metric 1
* 4000::/16, via nexthop 2001:1::1, metric 1
  5000::/16, interface VLAN2, metric 1
* 5555::/16, via nexthop 4000::1, metric 1
  5555::/16, via nexthop 9999::1, metric 1
* 5555::/16, via nexthop 4001:AF00::1, metric 1
* 6000::/16, via nexthop 2007::1, metric 1
```

Example 2. The following is sample output from the `show ipv6 route` command when entered with the IPv6 prefix `2001:200::/35`:

```
console# show ipv6 static 2001:200::/35
IPv6 Static routes   Code: * - installed in Routing Information
Base (RIB)
IPv6 Static routes distance is 1
* 2001:200::/35, via nexthop 4000::1, metric 1
  2001:200::/35, via nexthop 9999::1, metric 1
```

```
* 2001:200::/35, interface VLAN1, metric 1
```

Example 3. The following is sample output from the `show ipv6 route` command when entered with the interface VLAN 1:

```
console# show ipv6 static interface vlan 1  
IPv6 Static routes   Code: * - installed in Routing Information  
Base (RIB)  
IPv6 Static routes distance is 1  
* 5000::/16, interface VLAN1, metric 1
```

Example 4. The following is sample output from the `show ipv6 route` command with the `detail` keyword:

```
console# show ipv6 static detail  
IPv6 Static routes   Code: * - installed in Routing Information  
Base (RIB)  
IPv6 Static routes distance is 1  
* 3000::/16, interface VLAN1, metric 1  
* 4000::/16, via nexthop 2001:1::1, metric 1  
  5000::/16, interface fa 1/0/10, metric 1  
    Interface is down  
* 5555::/16, via nexthop 4000::1, metric 1  
  5555::/16, via nexthop 9999::1, metric 1  
    Route does not fully resolve  
* 5555::/16, via nexthop 4001:AF00::1, metric 1  
* 6000::/16, via nexthop 2007::1, metric 1
```


IPv6 Tunnel Commands

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 number.

Default Configuration

N/A

Command Mode

Global Configuration mode

Example

The following example enters the Interface Configuration (Tunnel) mode.

```
console(config)# interface tunnel 1
console(config-if)# tunnel source auto
console(config-if)# exit
```

tunnel isatap solicitation-interval

Use the `tunnel isatap solicitation-interval` Global Configuration mode command to set the time interval between unsolicited 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 unsolicited router solicitation messages sent to discover an ISATAP router.

Example

The following example sets the time interval between ISATAP router solicitation messages to 30 seconds.

```
console(config)# tunnel isatap solicitation-interval 30
```

tunnel isatap robustness

Use the `tunnel isatap robustness` Global Configuration mode command to configure the number of 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 router solicitation refresh messages that the device sends. (Range: 1–20).

Default Configuration

The default number of router solicitation refresh messages that the device sends is 3.

Command Mode

Global Configuration mode

User Guidelines

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 router solicitation refresh messages that the device sends to 5.

```
console(config)# tunnel isatap robustness 5
```

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 this router 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 (Tunnel) Configuration mode

User Guidelines

This 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.

The empty string means that automatic lookup is not applied.

Example

The following example configures the global string ISATAP2 as the automatic tunnel router domain name.

```
console(config)# interface tunnel 1
console(config-if)# tunnel isatap router ISATAP2
console(config-if)# exit
```

tunnel mode ipv6ip

Use the **tunnel mode ipv6ip** command in Interface Configuration mode to configure a static IPv6 tunnel interface. To remove an IPv6 tunnel interface, use the **no** form of this command.

Syntax

tunnel mode ipv6ip isatap

no tunnel mode ipv6ip

Parameters

- **isatap**—Specifies IPv6 automatic tunneling mode as ISATAP to connect IPv6 nodes (hosts and routers) within IPv4 networks.

Default Configuration

IPv6 tunnel interfaces are not configured.

Command Mode

Interface (Tunnel) Configuration mode

User Guidelines

IPv6 tunneling consists of encapsulating IPv6 packets within IPv4 packets for transmission across an IPv4 routing infrastructure.

The IPv6 interface is automatically enabled on a tunnel when it is configured as an IPv6 tunnel by the **tunnel mode ipv6ip** command and the local IPv4 is defined by the **tunnel source** command.

The IPv6 interface on an IPv6 tunnel is disabled if the tunnel stops to be an IPv6 tunnel or the tunnel local IPv4 address is removed and the new IPv4 cannot be chosen.

ISATAP Tunnels

Using this command with the **isatap** keyword specifies an automatic ISATAP tunnel. ISATAP tunnels enable transport of IPv6 packets within network boundaries. ISATAP tunnels allow individual IPv4/IPv6 dual-stack hosts within a site to connect to an IPv6 network using the IPv4 infrastructure.

ISATAP IPv6 addresses can use any initial Unicast /48 prefix. The final 64 bits are an interface identifier. Of these, the leading 32 bits are the fixed pattern 0000:5EFE; the last 32 bits carry the tunnel endpoint IPv4 address.

Only the **ipv6 address eui-64** command can be used to configured a global unicast IPv6 on an ISATAP tunnel.

Example

Example 1—The following example configures an ISATAP tunnel:

```
console(config)# interface vlan 1
console(config-if)# ip address 1.1.1.1 255.255.255.0
console(config-if)# exit
console(config)# interface tunnel 1
console(config-if)# tunnel mode ipv6ip isatap
console(config-if)# tunnel source 1.1.1.1
```

```
console(config-if)# ipv6 address 3ffe:b00:c18:1::/64 eui-64  
console(config-if)# exit
```

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 | interface-id}
```

```
no tunnel source
```

Parameters

- **auto**—The system minimum IPv4 address is used as the local IPv4 address (IPv4 address of the local tunnel endpoint).
- *ipv4-address*—Specifies the IPv4 address to use as the local IPv4 address (IPv4 address of the local tunnel endpoint).
- *interface-id*—Interface which the minimum IPv4 address is used as the local IPv4 address (IPv4 address of the local tunnel endpoint).

Default

No source address is defined.

Command Mode

Interface (Tunnel) Configuration mode

User Guidelines

If the **auto** or *interface-id* option is configured once time chosen IPv4 is used as the tunnel local IPv4 address until it is defined. A new IPv4 interface is only chosen in the following cases:

- After reboot.
- The used IPv4 is removed from the switch configuration.
- The tunnel mode is changed.

When the tunnel local IPv4 address is changed the IPv6 interface on the tunnel is re-enabled that causes removing static IPv6 configuration on the tunnel (for example, global IPv6 addresses, static IPv6 routes via the tunnel, etc.).

Example

```
console(config)# interface tunnel 1
console(config-if)# tunnel source 120.12.3.4
console(config-if)# exit
```

show ipv6 tunnel

Use the `show ipv6 tunnel EXEC` mode command to display information on the ISATAP tunnel.

Syntax

```
show ipv6 tunnel [all]
```

Parameters

- `all`—The switch displays all parameters of the tunnel. If the keyword is not configured only the tunnel parameters corresponding to its type are displayed.

Command Mode

User EXEC mode

Example

Example 1. The following example displays information on the ISATAP tunnel, when the `all` keyword is not configured:

```
console# show ipv6 tunnel
Tunnel 2
  Tunnel type           : ISATAP
  Tunnel status        : UP
  Tunnel Local address type : auto
```

```
Tunnel Local Ipv4 address      : 192.1.3.4
Router DNS name                 : ISATAP
Router IPv4 addresses
  1.1.1.1                      Detected
  100.1.1.1                   Detected
  14.1.100.1                  Not Detected
Router Solicitation interval    : 10 seconds
Robustness                      : 2
```

IP Routing Protocol-Independent Commands

ip route

To establish static routes, use the **ip route** command in global configuration mode. To remove static routes, use the **no** form of this command.

Syntax

```
ip route prefix {mask | /prefix-length} {{ip-address [metric value]}}
```

```
no ip route prefix {mask | /prefix-length} [ip-address]
```

Parameters

- *prefix*—IP route prefix for the destination.
- *mask*—Prefix mask for the destination.
- /*prefix-length*—Prefix mask for the destination. 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*—IP address of the next hop that can be used to reach that network.
- *metric value*—Metric of the route. The default metric is 1 for the Next Hop on an In-Band interface and 1 for the Next Hop on OOB. Range: 1–255.

Default Configuration

No static routes are established.

Command Mode

Global Configuration mode

User Guidelines

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.

Example

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(config)# 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(config)# 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(config)# 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(config)# 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(config)# no ip route 194.1.1.0 /24 1.1.1.1
```

ip routing

To enable IP routing, use the **ip routing** command in global configuration mode. To disable IP routing, use the **no** form of this command.

Syntax

`ip routing`

`no ip routing`

Parameters

This command has no arguments or keywords.

Default Configuration

IP routing is disabled.

Command Mode

Global Configuration mode

User Guidelines

Use the command to enable IP Routing.

The switch supports one IPv4 stack on in-band interfaces and the OOB port.

The IP stack is always running on the OOB port as an IP host regardless whether IP routing is enabled.

The switch blocks routing between in-band interfaces and the OOB interface.

In the case when there are two best routes - one via an in-band and one via the OOB port, the switch will use the route via the OOB port.

DHCP Relay and IP Helper cannot be enabled on the OOB port.

Example The following example enables IP routing

```
console(config)# ip routing
```

show ip route

To display the current state of the routing table, use the **show ip route** command in user EXEC or privileged EXEC mode.

Syntax

show ip route [*address ip-address* {*mask* [*longer-prefixes*]}] [*static* | *rejected* | *icmp* | *connected*]

Parameters

- **address ip-address**—IP address about which routing information should be displayed.
- **mask**—The value of the subnet mask.
- **longer-prefixes**—Specifies that only routes matching the IP address and mask pair should be displayed.
- **connected**—Displays connected routes.
- **icmp**—Displays routes added by ICMP Direct.
- **rejected**—Displays rejected routes.
- **static**—Displays static routes.

Command Mode

User EXEC mode

Privileged EXEC mode

User Guidelines

Use this command without parameters to display the whole IPv6 Routing table.

Use this command with parameters to specify required routes.

Examples

Example 1. The following is sample output from the **show ip route** command when IP Routing is not enabled:

```
console# show ip route
```

Maximum Parallel Paths: 1 (1 after reset)

IP Forwarding: disabled

Codes: > - best, C - connected, S - static, I - ICMP

IP Routing Table - 5 entries

Code	IP Route	Distance/ Metric	Next Hop IP Address	Last Time Updated	Outgoing Interface
S>	10.10.0.0/16	1/128	10.119.254.244	00:02:22	vlan2
S>	10.10.0.0/16	1/128	10.120.254.244	00:02:22	vlan3
S>	10.16.2.0/24	1/128	10.119.254.244	00:02:22	vlan2
C>	10.119.0.0/16	0/1	0.0.0.0		vlan2
C>	10.120.0.0/16	0/1	0.0.0.0		vlan3

Example 2. The following is sample output from the `show ip route` command when IP Routing is enabled:

console# `show ip route`

Maximum Parallel Paths: 1 (1 after reset)

IP Forwarding: enabled

Codes: > - best, C - connected, S - static

Codes: > - best, C - connected, S - static

IP Routing Table - 4 entries

Code	IP Route	Distance/ Metric	Next Hop IP Address	Last Time Updated	Outgoing Interface
C>	10.159.0.0/16	0/1	0.0.0.0		vlan2
C>	10.170.0.0/16	0/1	0.0.0.0		vlan2
S>	10.175.0.0/16	1/1	10.119.254.240		vlan2
S>	10.180.0.0/16	1/1	10.119.254.240		vlan2

Example 3. In the following example, the logical AND operation is performed on the address 10.16.0.0 and the mask 255.255.0.0, resulting in 10.16.0.0. On each destination in the routing table the logical AND operation is also

performed with the mask and the result is compared with 10.16.0.0. Any destinations that fall into that range are displayed in the output:

```
console# show ip route 10.16.0.0 255.255.0.0 longer-prefix
Maximum Parallel Paths: 1 (1 after reset)
IP Forwarding: enabled
Codes: > - best, C - connected, S - static
IP Routing Table - 6 entries
Code  IP Route    Distance/ Next Hop    Last Time Outgoing
      Metric   IP Address  Updated  Interface
-----
S>   10.16.2.0/24  110/128  10.119.254.244  00:02:22  vlan2
S>   10.16.2.64/26 110/128  100.1.14.244   00:02:22  vlan1
S>   10.16.2.128/26 110/128  110.9.2.2      00:02:22  vlan3
S>   10.16.208.0/24 110/128  120.120.5.44   00:02:22  vlan2
S>   10.16.223.0/24 110/128  20.1.2.24      00:02:22  vlan5
S>   10.16.236.0/24 110/129  30.19.54.240   00:02:23  vlan6
```

show ip route summary

Use the `show ip route summary` command in User EXEC or Privileged EXEC mode to display the current contents of the IP routing table in summary format.

Syntax

`show ip route summary`

Parameters

N/A.

Command Mode

User EXEC mode

Privileged EXEC mode

User Guidelines

Example

The following is sample output from the show **ip route summary** command:

```
console# show ip route summary
IP Routing Table Summary - 90 entries
35 connected, 25 static
Number of prefixes:
/16: 16, /18: 10, /22: 15, /24: 19
```


ACL Commands

ip access-list (IP extended)

Use the `ip access-list extended` 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 input` 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. (Range 1-32 characters)

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)#
```

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). Use the **no** form of the command to remove the access control entry.

Syntax

```
permit protocol {any / source source-wildcard} {any / destination destination-wildcard} [dscp number / precedence number] [time-range time-range-name]
```

```
[log-input]
```

```
permit icmp {any / source source-wildcard} {any / destination destination-wildcard} [any / icmp-type] [any / icmp-code]] [dscp number / precedence number] [time-range time-range-name]
```

```
[log-input]
```

```
permit igmp {any / source source-wildcard} {any / destination destination-wildcard}[igmp-type] [dscp number / precedence number] [time-range time-range-name]
```

```
[log-input]
```

```
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] [time-range time-range-name]
```

```
[log-input]
```

```
permit udp {any / source source-wildcard} {any / source-port/port-range} {any / destination destination-wildcard} {any / destination-port/port-range} [dscp number / precedence number] [time-range time-range-name]
```

```
[log-input]
```

```
no permit protocol {any / source source-wildcard} {any / destination destination-wildcard} [dscp number / precedence number][time-range time-range-name]
```

```
[log-input]
```

no permit icmp {*any* / *source source-wildcard*} {*any* / *destination destination-wildcard*} [*any* / *icmp-type*] [*any* / *icmp-code*]] [**dscp number** / **precedence number**]/**[time-range time-range-name]**

[log-input]

no permit igmp {*any* / *source source-wildcard*} {*any* / *destination destination-wildcard*}[*igmp-type*] [**dscp number** / **precedence number**]
[time-range time-range-name]

[log-input]

no 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**] **[time-range time-range-name]**

[log-input]

no permit udp {*any* / *source source-wildcard*} {*any*/ *source-port/port-range*} {*any* / *destination destination-wildcard*} {*any*/ *destination-port/port-range*} [**dscp number** / **precedence number**] **[time-range time-range-name]**

[log-input]

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.
- ***time-range-name***—Name of the time range that applies to this permit statement. (Range: 1–32)

- **log-input**—Specifies sending an informational SYSLOG message about the packet that matches the entry. Because forwarding/dropping is done in hardware and logging is done in software, if a large number of packets match an 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

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 any
```

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). Use the no form of the command to remove the access control entry.

Syntax

```
deny protocol {any | source source-wildcard} {any | destination destination-wildcard} [dscp number | precedence number] [time-range time-range-name] [disable-port |log-input ]
```

deny icmp {any / source source-wildcard} {any / destination destination-wildcard} [any / icmp-type] [any / icmp-code]] [dscp number / precedence number] [time-range time-range-name] [disable-port /log-input]

deny igmp {any / source source-wildcard} {any / destination destination-wildcard}[igmp-type] [dscp number / precedence number] [time-range time-range-name] [disable-port /log-input]

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][time-range time-range-name] [disable-port /log-input]

deny udp {any / source source-wildcard} {any/source-port/port-range} {any / destination destination-wildcard} {any/destination-port/port-range} [dscp number / precedence number] [time-range time-range-name] [disable-port /log-input]

no deny protocol {any / source source-wildcard} {any / destination destination-wildcard} [dscp number / precedence number] [time-range time-range-name] [disable-port /log-input]

no deny icmp {any / source source-wildcard} {any / destination destination-wildcard} [any / icmp-type] [any / icmp-code]] [dscp number / precedence number]/[time-range time-range-name] [disable-port /log-input]

no deny igmp {any / source source-wildcard} {any / destination destination-wildcard}[igmp-type] [dscp number / precedence number] [time-range time-range-name] [disable-port /log-input]

no 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] [time-range time-range-name] [disable-port /log-input]

no deny udp {any / source source-wildcard} {any/source-port/port-range} {any / destination destination-wildcard} {any/destination-port/port-range} [dscp number / precedence number] [time-range time-range-name] [disable-port /log-input]

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.
- **time-range-name**—Name of the time range that applies to this permit statement. (Range: 1–32)
- **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/dropping is done in hardware and logging is done in software, if a large number of packets match an 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

The number of TCP/UDP ranges that can be defined in ACLs is limited. 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-al)# deny ip 176.212.0.0 00.255.255 any
```

ipv6 access-list (IPv6 extended)

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 input](#) 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 1-32 characters.

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
```

permit (IPv6)

Use the **permit** command in Ipv6 Access-list Configuration mode to set permit conditions (ACEs) for IPv6 ACLs. Use the **no** form of the command to remove the access control entry.

Syntax

permit *protocol* {**any** | {*source-prefix/length*}}{**any** | *destination-prefix/length*} [*dscp number* | **precedence number**] [**time-range** *time-range-name*]

[*log-input*]

permit **icmp** {**any** | {*source-prefix/length*}}{**any** | *destination-prefix/length*} {**any**/*icmp-type*} {**any**/*icmp-code*} [*dscp number* | **precedence number**] [**time-range** *time-range-name*]

[*log-input*]

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*] [**time-range** *time-range-name*]

/log-input

permit *udp* {*any* | {*source-prefix/length*}} {*any* | *source-port/port-range*}} {*any* | *destination-prefix/length*} {*any* | *destination-port/port-range*} [*dscp number* | *precedence number*] [**time-range** *time-range-name*]

/log-input

no permit *protocol* {*any* | {*source-prefix/length*}} {*any* | *destination-prefix/length*} [*dscp number* | **precedence number**] [**time-range** *time-range-name*]

/log-input

no permit *icmp* {*any* | {*source-prefix/length*}} {*any* | *destination-prefix/length*} {*any* | *icmp-type*} {*any* | *icmp-code*} [*dscp number* | **precedence number**] [**time-range** *time-range-name*]

/log-input

no 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*] [**time-range** *time-range-name*]

/log-input

no permit *udp* {*any* | {*source-prefix/length*}} {*any* | *source-port/port-range*}} {*any* | *destination-prefix/length*} {*any* | *destination-port/port-range*} [*dscp number* | **precedence number**] [**time-range** *time-range-name*]

/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 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), dnsmx (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.
- ***time-range-name***—Name of the time range that applies to this permit statement. (Range: 1–32)

- **log-input**—Specifies sending an informational SYSLOG message about the packet that matches the entry. Because forwarding/dropping is done in hardware and logging is done in software, if a large number of packets match an 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

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.

The number of TCP/UDP ranges that can be defined in ACLs is limited. 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-acl)# permit tcp 3001::2/64 any any 80
```

deny (IPv6)

Use the **deny** command in IPv6 Access-list Configuration mode to set permit conditions (ACEs) for IPv6 ACLs. Use the no form of the command to remove the access control entry.

Syntax

deny protocol {*any* | {*source-prefix/length*}{*any* | *destination-prefix/length*}} [*dscp number* | *precedence number*] [*time-range time-range-name*] [*disable-port* | *log-input*]

deny icmp {*any* | {*source-prefix/length*}{*any* | *destination-prefix/length*}} {*any/icmp-type*} {*any/icmp-code*} [*dscp number* | *precedence number*] [*time-range time-range-name*] [*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*] [*time-range time-range-name*] [*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*] [*time-range time-range-name*] [*disable-port* | *log-input*]

no deny protocol {*any* | {*source-prefix/length*}} {*any* | *destination-prefix/length*}} [*dscp number* | *precedence number*] [*time-range time-range-name*] [*disable-port* | *log-input*]

no deny icmp {*any* | {*source-prefix/length*}} {*any* | *destination-prefix/length*}} {*any/icmp-type*} {*any/icmp-code*} [*dscp number* | *precedence number*] [*time-range time-range-name*] [*disable-port* | *log-input*]

no 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*] [*time-range time-range-name*] [*disable-port* | *log-input*]

no deny udp {*any* | {*source-prefix/length*}} {*any* | *source-port/port-range*}} {*any* | *destination-prefix/length*}} {*any* | *destination-port/port-range*}} [*dscp number* | *precedence number*] [*time-range time-range-name*] [*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.
- **time-range-name**—Name of the time range that applies to this permit statement. (Range: 1–32)
- **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/dropping is done in hardware and logging is done in software, if a large number of packets match an 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. 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
```

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 input** 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: 1–32 characters).

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
```

permit (MAC)

Use the **permit** command in MAC Access-list Configuration mode to set permit conditions (ACEs) for a MAC ACL. Use the **no** form of the command to remove the access control entry.

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] [time-range time-range-name]
```

```
[log-input]
```

```
no 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] [time-range time-range-name]
```

```
[log-input]
```

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.
- **time-range-name**—Name of the time range that applies to this permit statement. (Range: 1–32)

- **log-input**—Specifies sending an informational SYSLOG message about the packet that matches the entry. Because forwarding/dropping is done in hardware and logging is done in software, if a large number of packets match an 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

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
```

deny (MAC)

Use the **deny** command in MAC Access-list Configuration mode to set deny conditions (ACEs) for a MAC ACL. Use the **no** form of the command to remove the access control entry.

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*] [*time-range time-range-name*] [*disable-port* | *log-input*]

no 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*] [*time-range time-range-name*] [*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.
- *time-range-name*—Name of the time range that applies to this permit statement. (Range: 1–32)
- *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/dropping is done in hardware and logging is done in software, if a large number of packets match an 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

Example

```
console(config)# mac access-list extended server1
console(config-mac-acl)# deny 00:00:00:00:00:01 00:00:00:00:00:ff
any
```

service-acl input

Use the **service-acl input** 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] [default-action {deny-any | permit-any}]
```

```
no service-acl input
```

Parameters

- **acl-name**—Specifies an ACL to apply to the interface. See the user guidelines. (Range: 1–32 characters).
- **deny-any**—Deny all packets (that were ingress at the port) that do not meet the rules in this ACL.
- **permit-any**—Forward all packets (that were ingress at the port) that do not meet the rules in this ACL.

Default Configuration

No ACL is assigned.

Command Mode

Interface Configuration mode (Ethernet, Port-Channel,)

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.
- When the user binds ACL to an interface, TCAM resources will be consumed. One TCAM rule for each MAC or IP ACE and two TCAM rules for each IPv6 ACE. The TCAM consumption is always even number, so in case of odd number of rules the consumption will be increased by 1.

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 te0/1
console(config-if)# service-acl input server-acl default-
action deny-any
```

service-acl input block

Use the `service-acl input` Interface Configuration mode commands to discard packets that are classified to specific protocols. Use the `no` form of those commands to disable discarding of the packets.

Syntax

```
service-acl input protocol1 [protocol2 ... protocol6]
```

```
no service-acl input
```

Parameters

protocol—Specifies a protocol to filter. Available values are: `blockcdp`, `blockvtp`, `blockdtp`, `blockudld`, `blockpagp`, `blocksstp`, and `blockall`.

Default Configuration

No protocol is defined.

Command Mode

Interface Configuration mode (Ethernet, Port-Channel)

User Guidelines

If you want to define multiple protocols on the same interface, those protocols should be defined in the same command.

To change configuration of the protocol filtering for an interface, you should first remove the current assignment of protocol filtering assignment, and then assign the new configuration of the protocol filtering.

If Proprietary Protocol Filtering rules are assigned on an interface, the user is not able to assign ACL or Policy Map or Security suite rules to that interface and to enable 802.IX Dynamic Policy Assignment to that interface.

If ACL or Policy Map or Security suite rules are assigned to an interface or 802.IX Dynamic Policy Assignment is enabled for an interface, the user is not able to assign Proprietary Protocol Filtering rules to that interface.

The following table defines the DA and protocol types of the packets that are subject for discarding per each command:

Command	Destination Address	Protocol Type
blockcdp	0100.0ccc.cccc	0x2000
blockvtp	0100.0ccc.cccc	0x2003
blockdtp	0100.0ccc.cccc	0x2004
blockudld	0100.0ccc.cccc	0x0111
blockpagp	0100.0ccc.cccc	0x0104
blocksstp	0100.0ccc.cccd	-
blockall	0100.0ccc.ccc0 - 0100.0ccc.cccf	-

Example

```
console(config-if)# service-acl input blockcdp blockvtp
```

service-acl output

Use the **service-acl output** command in Interface Configuration mode to control access to an interface on the egress (transmit path).

Use the **no** form of this command to remove the access control.

Syntax

```
service-acl output acl-name1 [acl-name2]
```

```
no service-acl output
```

Parameters

acl-name-Specifies an ACL to apply to the interface. See the user guidelines. (Range: acl-name is from 0-32 characters. Use "" for empty string)

Default

No ACL is assigned.

Command Mode

Interface Configuration mode(Ethernet, Port-Channel).

User Guidelines

The rule actions: log-input is not supported. Trying to use it will result in an error.

The deny rule action disable-port is not supported. Trying to use it will result in an error.

IPv4 and IPv6 ACLs can be bound together on an interface.

A MAC ACL cannot be bound on an interface together with an IPv4 ACL or IPv6 ACL.

Two ACLs of the same type cannot be added to a port.

An ACL cannot be added to a port that is already bounded to an ACL, without first removing the current ACL and binding the two ACLs together.

Example

This example binds an egress ACL to a port:

```
console(config)# mac access-list extended server
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 te0/1
console(config-if)# service-acl output server
```

time-range

Use the **time-range** Global Configuration mode command to define time ranges for different functions. In addition, this command enters the Time-range Configuration mode. All commands after this one refer to the time-range being defined.

This command sets a time-range name. Use the [absolute](#) and [periodic](#) commands to actually configure the time-range.

Use the **no** form of this command to remove the time range from the device.

Syntax

time-range *time-range-name*

no time-range *time-range-name*

Parameters

time-range-name—Specifies the name for the time range. (Range: 1–32 characters)

Default Configuration

No time range is defined

Command Mode

Global Configuration mode

User Guidelines

After adding the name of a time range with this command, use the [absolute](#) and [periodic](#) commands to actually configure the time-range. Multiple periodic commands are allowed in a time range. Only one absolute command is allowed.

If a time-range command has both absolute and periodic values specified, then the periodic items are evaluated only after the absolute start time is reached, and are not evaluated again after the absolute end time is reached.

All time specifications are interpreted as local time.

To ensure that the time range entries take effect at the desired times, the software clock should be set by the user or by SNTP. If the software clock is not set by the user or by SNTP, the time range ACEs are not activated.

The user cannot delete a time-range that is bound to any features.

When a time range is defined, it can be used in the following commands:

- dot1x port-control
- power inline
- operation time
- permit (IP)
- deny (IP)
- permit (IPv6)
- deny (IPv6)
- permit (MAC)
- deny (MAC)

Example

```
console(config)# time-range http-allowed  
console(config-time-range)#periodic mon 12:00 to wed 12:00
```

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 http-allowed
```

```
console(config-time-range)# absolute start 12:00 1 jan 2005
console(config-time-range)# absolute end 12:00 31 dec 2005
```

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 *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: mon, tue, wed, thu, fri, sat, and sun.
- **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 http-allowed
console(config-time-range)# periodic mon 12:00 to wed 12:00
```

show time-range

Use the `show time-range` User EXEC mode 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

User EXEC mode

Example

```
console# show time-range
http-allowed
```

```
absolute start 12:00 1 Jan 2005 end 12:00 31 Dec 2005
```

```
periodic Monday 12:00 to Wednesday 12:00
```

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 time-range-active [name]
```

Parameters

- **name**—Specifies the name of the ACL. (Range: 1-160 characters).
- **time-range-active**—Shows only the Access Control Entries (ACEs) whose time-range is currently active (including those that are not associated with time-range).

Command Mode

Privileged EXEC mode

Example

```
console# show access-lists
```

```
Standard IP access list 1
```

```
Extended IP access list ACL2
```

```
permit 234 172.30.19.1 0.0.0.255 any time-range weekdays
```

```
permit 234 172.30.23.8 0.0.0.255 any time-range weekdays
```

```
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
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
```

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          ACLs
-----          -
te0/1              blockcdp, blockvtp
te0/2              Ingress: server1
                   Egress : ip
```

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 te0/1
```

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
```

```
-----
```

```
-----
```

```
te0/1              79
```

```
te0/2              9
```

```
te0/3              0
```

```
Number of hits that were counted in global counter (due to lack of  
resources) =19
```

Quality of Service (QoS) Commands

qos

Use the **qos** Global Configuration mode command to enable QoS on the device and set its mode. Use the **no** form of this command to disable QoS on the device.

Syntax

qos [*basic* | *advanced*]

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.

Default Configuration

If **qos** is entered without any keywords, the QoS **basic** mode is **enabled**.

Command Mode

Global Configuration mode

Examples

Example 1—The following example enables QoS basic mode on the device.

```
console(config)# qos
```

Example 2—The following example enables QoS advanced mode on the device with the **ports-not-trusted** option.

```
console(config)# qos advanced
```

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

Privileged EXEC mode

User Guidelines

Trust mode is displayed if QoS is enabled in basic mode.

Examples

```
console(config)# show qos
Qos: Disabled
console(config)# show qos
Qos: Basic mode
Basic trust: dscp
console(config)# show qos
Qos: Advanced mode
Advanced mode trust type: cos
Advanced mode ports state: Trusted
```

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
```

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

Privileged EXEC mode

Example

The following example displays the class map for Class1.

```
console(config)# show class-map  
Class Map matchAny class1  
    Match access-group mac
```

match

Use the **match** Class-map Configuration mode command to bind the 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
```

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)#
```

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
```

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

Privileged EXEC mode

Example

The following example displays all policy maps.

```
console(config)# 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
```

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

No trust state

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.

Trust values set with this command supersede trust values set on specific interfaces with the `qos trust (Interface)` Interface Configuration mode command.

The `trust` and `set` commands are mutually exclusive within the same policy map.

Policy maps, which contain `set` or `trust` commands or that have ACL classification to an egress interface, cannot be attached 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-ip-1)# permit ip any any
console(config-ip-1)# 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 cos-dscp
```

set

Use the **set** Policy-map Class Configuration mode command to select the value that QoS uses as the DSCP value, the egress queue or to set 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: 1-8)
- **cos** *new-cos*—Specifies the new user priority to be marked in the packet. (Range: 0–7)

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 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-ip-1)# permit ip any any
console(config-ip-1)# exit
```

```
console(config)# class-map c1
console(config-cmap)# match access-group ipl
console(config-cmap)# exit
console(config)# policy-map p1
console(config-pmap)# class c1
console(config-pmap-c)# set dscp 56
```

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: 3–maximal port speed)
- **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

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)#
console(config-pmap-c)# police 124000 9600 exceed-action drop
```

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 (Ethernet) Configuration mode

User Guidelines

Only one policy map per interface per direction is supported.

Example

Example 1—The following example attaches a policy map called Policy1 to the input interface.

```
console(config-if)# service-policy input policy1
```

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–maximal port speed)
- `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

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
```

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

Privileged EXEC mode

Example

The following example displays the parameters of the aggregate policer called `Policer1`.

```
console(config)# show qos aggregate-policer policer1
aggregate-policer policer1 96000 4800 exceed-action drop
not used by any policy map
```

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 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
```

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 8 queues is as follows:

CoS value 0 is mapped to queue 1.

CoS value 1 is mapped to queue 2.

CoS value 2 is mapped to queue 3.

CoS value 3 is mapped to queue 6.

CoS value 4 is mapped to queue 5.

CoS value 5 is mapped to queue 8.

CoS value 6 is mapped to queue 8

CoS value 7 is mapped to queue 7

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
```

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... weighting
```

```
no wrr-queue bandwidth
```

Parameters

`weight1 weight1... weighting` 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 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)# priority-queue out num-of-queues 0
console(config)# wrr-queue bandwidth 6 6 6 6 6 6 6 6
```

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: 0–8 .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** = 8 , all the 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
```

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: ,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 (Ethernet) Configuration mode

Example

The following example sets a traffic shaper on te0/1 when the average traffic rate exceeds 64 kbps or the normal burst size exceeds 4096 bytes.

```
console(config)# interface te0/1
console(config-if)# traffic-shape 64 4096
```

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: 1-8).
- **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 (Ethernet) Configuration mode

Example

The following example sets a shaper on queue 1 on te0/1 when the average traffic rate exceeds 124000 kbps or the normal burst size exceeds 9600 bytes.

```
console(config)# interface te0/1
console(config-if)# traffic-shape queue 1 64 4096
```

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 3–maximal port speed.
- **burst committed-burst-bytes**—The burst size in bytes. (Range: 3000–19173960). If unspecified, defaults to 128K.

Default Configuration

Rate limiting is disabled.

Committed-burst-bytes is 128K.

Command Mode

Interface (Ethernet) Configuration 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 te0/1 to 150,000 kbps.

```
console(config)# interface te0/1
console(config-if)# rate-limit 150000
```

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(config)# qos wrr-queue wrtd
This setting will take effect only after copying running configuration
to startup configuration and resetting the device
console(config)#
```

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

Privileged EXEC mode

Example

```
console(config)# show qos wrr-queue wrtd
Weighted Random Tail Drop is disabled
Weighted Random Tail Drop will be enabled after reset
```

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 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

Privileged 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.

In case of Policers, Shapers and Rate Limit - only the ports which are not in the default configuration will be showed.

Examples

Example 1—The following is an example of the output from the **show qos interface queueing** command for 4 queues.

```
Ethernet te0/0/1
wrr bandwidth weights and EF priority:
qid-weights      Ef - Priority
1 - N/A          ena- 1
2 - N/A          ena- 2
3 - N/A          ena- 3
```

```
4 - N/A          ena- 4
Cos-queue map:
cos-qid
0 - 1
1 - 1
2 - 2
3 - 3
4 - 3
5 - 4
6 - 4
7 - 4
```

Example 2 —The following an example of the output from the `show qos interface buffers` command for 8 queues

```
console(config)# show qos interface buffers te0/1
te0/1
Notify Q depth:
buffers te0/1
Ethernet te0/1
qid  thresh0  thresh1  thresh2
1    100      100      80
2    100      100      80
3    100      100      80
4    100      100      80
5    100      100      80
6    100      100      80
7    100      100      80
8    100      100      80
```

Example 3—This is an example of the output from the `show qos interface shapers` command.

```
console(config)# show qos interface shapers te0/1
te0/1
Port shaper: enable
Committed rate: 192000 bps
Committed burst: 9600 bytes
```

QID	Status	Target Committed Rate [bps]	Target Committed Burst [bytes]
1	Enable	100000	17000
2	Disable	N/A	N/A
3	Enable	200000	19000
4	Disable	N/A	N/A
5	Disable	N/A	N/A
6	Disable	N/A	N/A
7	Enable	178000	8000
8	Enable	23000	1000

Example 4—This is an example of the output from `show qos interface policer`

```
console(config)# show qos interface policer te0/1
Ethernet te0/1
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
```

Example 5—This is an example of the output from `show qos interface rate-limit`

```
console(config)# show qos interface rate-limit te0/1
Port rate-limit [kbps] Burst [Bytes]
```

```
-----
te0/1      3000      3000
```

```
console(config)# show qos interface rate-limit te0/1
Port          rate-limit [kbps]      Burst [Bytes]
-----
te0/1         1000                   512
```

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 tengigabitethernet 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.
- `tengigabitethernet`—Specifies that the thresholds are to be applied to 10 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 te0/1 80
```

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.

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
```

qos map dscp-queue

Use the `qos map dscp-queue` Global Configuration mode command to configure the DSCP to queue 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 8 queues is as follows.

DSCP value	0	1-8	9-15	16,24,40, 48-63	17-23	25-31	33-39	32,41-47
Queue-ID	2	1	3	7	4	5	6	8

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
```

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
```

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

CoS 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 QoS domain are classified at its edge. 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
```

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

Parameters

N/A

Default Configuration

Each port is enabled while the system is in basic mode.

Command Mode

Interface (Ethernet) Configuration mode

Example

The following example configures te0/1 to the default trust state.

```
console(config)# interface te0/1
console(config-if)# qos trust
```

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: 0–7)

Default Configuration

The default CoS value of a port is 0.

Command Mode

Interface (Ethernet) Configuration 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 `te0/1` default CoS value as 3.

```
console(config)# interface te0/1
console(config-if)# qos cos 3
```

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

Parameters

N/A

Default Configuration

Disabled

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
```

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
```

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.

Default Configuration

Display all maps.

Command Mode

Privileged EXEC mode

Example

The following example displays the QoS mapping information

```
console(config)# 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
  1 :  01 01 01 01 01 01 02 02 02 02
  2 :  02 02 02 02 02 02 02 02 02 02
  3 :  02 02 03 03 03 03 03 03 03 03
  4 :  03 03 03 03 03 03 03 03 04 04
  5 :  04 04 04 04 04 04 04 04 04 04
  6 :  04 04 04 04
```

clear qos statistics

Use the `clear qos statistics EXEC` mode command to clear the QoS statistics counters.

Syntax

```
clear qos statistics
```

Parameters

N/A

Default Configuration

N/A

Command Mode

Privileged EXEC mode

Example

The following example clears the QoS statistics counters.

```
console(config)# clear qos statistics
```

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.

This command is relevant only when policers are defined.

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 (Ethernet) Configuration mode

Example

The following example enables counting in-profile and out-of-profile on the interface.

```
console(config)# interface te0/1
console(config-if)# qos statistics policer policy1 class1
```

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
```

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
```

show qos statistics

Use the `show qos statistics EXEC` mode command to display Quality of Service statistical information.

Syntax

`show qos statistics`

Parameters

N/A

Default Configuration

N/A

Command Mode

Privileged 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(config)# show qos statistics
Policers
-----

Interface   Policy map   Class Map   In-profile   Out-of-profile bytes
-----
            -----
            -----
            bytes
            -----

te0/1       Policy1      Class1      7564575      5433
te0/1       Policy1      Class2      8759         52
te0/2       Policy1      Class1      746587458   3214
te0/2       Policy1      Class2      5326         23

Aggregate Policers
-----

Name        In-profile bytes   Out-of-profile bytes
-----
Policer1    7985687           121322

Output Queues
-----

Interface   Queue        DP          Total packets   TD packets
-----
            -----
            -----
            -----
            -----

te0/1       2            High       799921          1.2%
te0/2       All          High       5387326         0.2%
```


DNS Client Commands

clear host

Use the **clear host** command in privileged EXEC mode to delete dynamic hostname-to-address mapping entries from the DNS client name-to-address cache.

Syntax

```
clear host {hostname / *}
```

Parameters

- *hostname*—Name of the host for which hostname-to-address mappings are to be deleted from the DNS client name-to-address cache.
- *—Specifies that all the dynamic hostname-to-address mappings are to be deleted from the DNS client name-to-address cache.

Default Configuration

No hostname-to-address mapping entries are deleted from the DNS client name-to-address cache.

Command Mode

Privileged EXEC mode

User Guidelines

To remove the dynamic entry that provides mapping information for a single hostname, use the *hostname* argument. To remove all the dynamic entries, use the * keyword.

To define a static hostname-to-address mappings in the DNS hostname cache, use the **ip host** command.

To delete a static hostname-to-address mappings in the DNS hostname cache, use the **no ip host** command.

Example

The following example deletes all dynamic entries from the DNS client name-to-address cache.

```
console# clear host *
```

ip domain lookup

Use the **ip domain lookup** command in Global Configuration mode to enable the IP Domain Naming System (DNS)-based host name-to-address translation.

To disable the DNS, use the **no** form of this command.

Syntax

ip domain lookup

no ip domain lookup

Parameters

N/A

Default Configuration

Enabled.

Command Mode

Global Configuration mode

Example

The following example enables DNS-based host name-to-address translation.

```
console(config)# ip domain lookup
```

ip domain name

Use the **ip domain name** command in Global Configuration mode. to define a default domain name that the switch uses to complete unqualified hostnames (names without a dotted-decimal domain name).

To delete the static defined default domain name, use the **no** form of this command.

Syntax

ip domain name *name*

no ip domain name

Parameters

name—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 of each domain level is 63 characters.

Default Configuration

No default domain name is defined.

Command Mode

Global Configuration mode

User Guidelines

Any IP hostname that does not contain a domain name (that is, any name without a dot) will have the dot and the default domain name appended to it before being added to the host table.

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 each domain level 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 website.com
```

ip domain polling-interval

Use the **ip domain polling-interval** command in Global Configuration mode to specify the polling interval.

Use the **no** form of this command to return to the default behavior.

Syntax

ip domain polling-interval *seconds*

no ip domain polling-interval

Parameters

seconds—Polling interval in seconds. The range is from $(2 * (R+1) * T)$ to 3600.

Default Configuration

The default value is $2 * (R+1) * T$, where

- R is a value configured by the **ip domain retry** command.
- T is a value configured by the **ip domain timeout** command.

Command Mode

Global Configuration mode

User Guidelines

Some applications communicate with the given IP address continuously. DNS clients for such applications, which have not received resolution of the IP address or have not detected a DNS server using a fixed number of retransmissions, return an error to the application and continue to send DNS Request messages for the IP address using the polling interval.

Example

The following example shows how to configure the polling interval of 100 seconds:

```
console(config)# ip domain polling-interval 100
```

ip domain retry

Use the **ip domain retry** command in Global Configuration mode to specify the number of times the device will send Domain Name System (DNS) queries when there is no replay.

To return to the default behavior, use the **no** form of this command.

Syntax

ip domain retry *number*

no ip domain retry

Parameters

number—Number of times to retry sending a DNS query to the DNS server. The range is from 0 to 16.

Default Configuration

The default value is 1.

Command Mode

Global Configuration mode

User Guidelines

The number argument specifies how many times the DNS query will be sent to a DNS server until the switch decides that the DNS server does not exist.

Example

The following example shows how to configure the switch to send out 10 DNS queries before giving up:

```
console(config)# ip domain retry 10
```

ip domain timeout

Use the **ip domain timeout** command in Global Configuration mode to specify the amount of time to wait for a response to a DNS query.

To return to the default behavior, use the **no** form of this command.

Syntax

`ip domain timeout seconds`
`no ip domain timeout`

Parameters

seconds—Time, in seconds, to wait for a response to a DNS query. The range is from 1 to 60.

Default Configuration

The default value is 2 seconds.

Command Mode

Global Configuration mode

User Guidelines

Use the command to change the default time out value. Use the **no** form of this command to return to the default time out value.

Example

The following example shows how to configure the switch to wait 50 seconds for a response to a DNS query:

```
console(config)# ip domain timeout 50
```

ip host

Use the **ip host** Global Configuration mode command to define the static host name-to-address mapping in the DNS host name cache.

Use the **no** form of this command to remove the static host name-to-address mapping.

Syntax

`ip host hostname address1 [address2...address8]`
`no ip host name ip host name [address1...address8]`

Parameters

- *hostname*—Name of the host. (Length: 1–158 characters. Maximum label length of each domain level is 63 characters).
- *address1*—Associated host IP address (IPv4 or IPv6, if IPv6 stack is supported).
- *address2...address8*—Up to seven additional associated IP addresses, delimited by a single space (IPv4 or IPv6, if IPv6 stack is supported).

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.

An IP application will receive the IP addresses in the following order:

1. IPv6 addresses in the order specified by the command.
2. IPv4 addresses in the order specified by the command.

Use the **no** format of the command with the *address1...address8* argument to delete the specified addresses. The entry is deleted if all its addresses are deleted.

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
```

ip name-server

Use the **ip name-server** command in Global Configuration mode to specify the address of one or more name servers to use for name and address resolution.

Use the **no** form of this command to remove the static specified addresses.

Syntax

```
ip name-server server1-address [server-address2...erver-address8]
```

```
no ip name-server [server-address1...server-address8]
```

Parameters

- *server-address1*—IPv4 or IPv6 addresses of a single name server.
- *server-address2...server-address8*—IPv4 or IPv6 addresses of additional name servers.

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.

Each **ip name-server** command replaces the configuration defined by the previous one (if one existed).

Example

The following example shows how to specify IPv4 hosts 172.16.1.111, 172.16.1.2, and IPv6 host 2001:0DB8::3 as the name servers:

```
console(config)# ip name-server 172.16.1.111 172.16.1.2  
2001:0DB8::3
```

show hosts

Use the **show hosts** command in privileged EXEC mode to display the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses.

Syntax

```
show hosts [all | hostname]
```

Parameters

- **all**—The specified host name cache information is to be displayed for all configured DNS views. This is the default.
- ***hostname***—The specified host name cache information displayed is to be limited to entries for a particular host name.

Command Mode

Privileged EXEC mode

Default Configuration

Default is **all**.

User Guidelines

This command displays the default domain name, a list of name server hosts, and the cached list of host names and addresses.

Example

The following is sample output with no parameters specified:

```
console# show hosts
Name/address lookup is enabled
Domain Timeout: 3 seconds
Domain Retry: 4 times
Domain Polling Interval: 10 seconds
Default Domain Table
Source  Interface Preference Domain
```

```

static                               website.com
dhcpv6  vlan 100      1      qqtca.com
dhcpv6  vlan 100      2      company.com
dhcpv6  vlan 1100     1      pptca.com

```

Name Server Table

Source	Interface	Preference	IP Address
static		1	192.0.2.204
static		2	192.0.2.205
static		3	192.0.2.105
DHCPv6	vlan 100	1	2002:0:22AC::11:231A:0BB4
DHCPv4	vlan 1	1	192.1.122.20
DHCPv4	vlan 1	2	154.1.122.20

Casche Table

Flags: (static/dynamic, OK/Ne/??)

OK - Okay, Ne - Negative Cache, ?? - No Response

Host Flag Address;Age...in preference order

```

example1.company.com (dynamic, OK) 2002:0:130F::0A0:1504:0BB4;1
112.0.2.10 176.16.8.8;123 124 173.0.2.30;39
example2.company.com (dynamic, ??)
example3.company.com (static, OK) 120.0.2.27
example4.company.com (dynamic, OK) 24 173.0.2.30;15
example5.company.com (dynamic, Ne); 12

```

UDLD Commands

show udld

To display the administrative and operational Unidirectional Link Detection Protocol (UDLD) status, use the **show udld** command in Privileged EXEC mode.

Syntax

```
show udld [interface-id] [neighbors]
```

Parameters

- *interface-id*—Interface identifier of an Ethernet port.
- *neighbors*—Displays neighbor information only.

Command Mode

Privileged EXEC mode

User Guidelines

If you do not enter an interface ID value, the administrative and operational UDLD status for all interfaces on which UDLD is enabled are displayed.

Example

Example 1—This example shows how to display the UDLD state for all interfaces. Most of the fields shown in the display are self-explanatory. Those that are not self-explanatory are defined below.

```
console# show udld
Global UDLD mode: normal
Message Time: 15 sec(default)
Interface te0/1
  Port UDLD mode: aggressive
  Port Current state: Bidirectional
  Number of detected neighbors: 1
```

Port Neighbor Table

```
Neighbor Device ID: 1234567893
  Neighbor MAC: 00:00:01:22:33:dd
  Neighbor Device name: switch A
  Neighbor Port ID: gil/2/1
  Neighbor Message Time: 20 sec
  Neighbor Current State: Bidirectional
  Neighbor Expiration Time: 7 sec
Neighbor Device ID: 1234544893
  Neighbor MAC: 00:00:01:22:33:ff
  Neighbor Device name: switch A
  Neighbor Port ID: gil/2/1
  Neighbor Message Time: 15 sec
  Neighbor Current State: Undetermined
  Neighbor Expiration Time: 17 sec
```

Interface te0/2

```
Port UDLD mode: normal (default)
Port Current state: Undetermined
Number of detected neighbors: 1
  Neighbor Device ID: 1234567753
    Neighbor MAC: 00:00:01:22:33:fe
    Neighbor Device name: switch A
    Neighbor Port ID: gil/2/1
    Neighbor Message Time: 15 sec
    Neighbor Current State: Undetermined
    Neighbor Expiration Time: 11 sec
```

Interface te0/3

```
Port UDLD mode: disabled
```

Interface te0/4

```
Port UDLD mode: normal (default)
Port Current state: shutdown
```

Field Descriptions:

- **Global UDLD mode**—The global UDLD mode (normal or aggressive) configured by the `udld` command.
- **Message Time**—The message time configured by the `udld message time` command.
- **Port UDLD mode**—The interface UDLD mode (normal or aggressive).

- **Port Current state**—The UDLD operational state: interface UDLD mode (normal or aggressive).
 - **Disabled**—UDLD is disabled on the port by the `udld port disable` command.
 - **Shutdown**—UDLD is enabled on the port and the port operational state is DOWN.
 - **Detection**—UDLD is detecting the link state.
 - **Bidirectional**—The link is bidirectional.
 - **Undetermined**—The link state is undetermined - no UDLD message has been received on the port.
- **Neighbor Device ID**—The device ID of the neighbor.
- **Neighbor MAC**—The MAC address of the neighbor.
- **Neighbor Device name**—The Device name of the neighbor.
- **Neighbor Port ID**—The device port ID of the neighbor on which the recent UDLD message was sent.
- **Neighbor Message Time**—The message time of the neighbor.
- **Neighbor Current State**—The current state of the neighbor:
 - **Bidirectional**—The UDLD messages received from the neighbor contain the Device ID and Port ID of the switch in the Echo TLV.
 - **Undetermined**—The UDLD messages received from the neighbor do not contain the Device ID and Port ID of the switch in the Echo TLV.
- **Neighbor Expiration Time**—Left time in seconds until the current neighbor state expires.

Example 2—This example shows how to display the UDLD state for one given interface:

```

console# show udld te0/1
Global UDLD mode: normal
Message Time: 15 sec(default)
Interface te0/1
  Port UDLD mode: aggressive
  Port Current state: Bidirectional

```

```

Number of detected neighbors: 1
Port Neighbor Table
Neighbor Device ID: 1234567893
  Neighbor MAC: 00:00:01:22:33:dd
  Neighbor Device name: switch A
  Neighbor Port ID: gil/2/1
  Neighbor Message Time: 20 sec
  Neighbor Current State: Bidirectional
  Neighbor Expiration Time: 7 sec
Neighbor Device ID: 1234544893
  Neighbor MAC: 00:00:01:22:33:ff
  Neighbor Device name: switch A
  Neighbor Port ID: gil/2/1
  Neighbor Message Time: 15 sec
  Neighbor Current State: Undetermined
  Neighbor Expiration Time: 17 sec

```

Example 3—This example shows how to display neighbor information only:

```

console# show udld neighbors

```

Port	Device ID	Port-ID	Device Name	Message	Neighbor Expiration	Time
(sec)				Time(sec)	State	
te0/1	1234567893	gil/0/1	SAL0734K5R2	15	Bidirect	11
te0/2	3456750193	gil/0/2	SAL0734K5R3	20	Undetermined	5

Example 4—This example shows how to display neighbor information only for a single interface:

```
console# show uddl te0/1 neighbors
```

Port	Device ID	Port-ID	Device Name	Message	Neighbor Expiration
(sec)				Time(sec) State	Time

te0/1	1234567893	gi1/0/1	SAL0734K5R2	15 Bidirect	11

uddl

Use the **uddl** command in Global Configuration mode to globally enable the UniDirectional Link Detection (UDLD) protocol. To disable UDLD, use the **no** form of this command.

Syntax

```
uddl aggressive | normal
```

```
no uddl
```

Parameters

- **aggressive**—Enables UDLD in aggressive mode by default on all fiber interfaces.
- **normal**—Enables UDLD in normal mode by default on all fiber interfaces.

Default Configuration

UDLD is disabled on all fiber interfaces.

Command Mode

Global Configuration mode

Interface (Ethernet) Configuration mode

User Guidelines

This command affects fiber interfaces only. Use the `udld port` command in Interface Configuration mode to enable UDLD on other interface types.

Use the `no` form of this command to disable UDLD on all fiber ports.

The device supports the UDLD protocol specified by RFC 5171.

UDLD supports two modes of operation: normal and aggressive. In the aggressive mode the device shuts down a port if it cannot explicitly detect that the link is bidirectional. In the normal mode the device shuts down an interface if it explicitly detect that the link is unidirectional. A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

You can use the `set interface active` command to reset an interface shut down by UDLD.

Example

This example shows how to enable UDLD on all fiber interfaces:

```
console(config)# udld normal
```

udld message timeout

Use the `udld message time` command in Global Configuration mode to configure a global value of the interval between two sent probe messages. To return to the default value, use the `no` form of this command.

Syntax

```
udld message timeout seconds
```

```
no udld message time
```

Parameters

seconds—Interval between two sent probe messages. The valid values are from 1 to 90 seconds.

Default Configuration

15 seconds.

Command Mode

Global Configuration mode

User Guidelines

Use this command to change the default value of the message interval - the interval between two sequential sent probe messages.

Example

This example shows how to globally set the interval to 40sec:

```
console(config)# udld message timeout 40
```

udld port

To enable the UDLD protocol on an Ethernet port, use the **udld port** command in Interface Configuration mode. To return to the default, use the **no** form of this command.

Syntax

```
udld port [aggressive | normal | disable]
```

```
no udld port
```

Parameters

- **aggressive**—Enables UDLD in aggressive mode on this interface.
- **normal**—Enables UDLD in normal mode on this interface. The normal keyword is applied if no keyword is specified.
- **disable**—Disables UDLD on this interface.

Default Configuration

The defaults are as follows:

- Fiber interfaces are in the state configured by the **udld** command.
- Non-fiber interfaces are in the Disable state.

Command Mode

Interface (Ethernet) Configuration mode

User Guidelines

Use this command on fiber ports to override the setting of the global **udld** command.

If the port changes from fiber to non-fiber or vice versa, all configurations are maintained because the platform software detects a change of module or a Gigabit Interface Converter (GBIC) change.

Examples

Example 1—This example shows how to enable UDLD in normal mode on an Ethernet port regardless of the current global **udld** setting:

```
console(config)# interface te0/1
console(config-if)# udld port normal
console(config-if)# exit
```

Example 2—This example shows how to return to the default configuration:

```
console(config)# interface te0/1
console(config-if)# no udld port
console(config-if)# exit
```

Example 3—This example shows how to disable UDLD on an Ethernet port regardless of the current global **udld** setting:

```
console(config)# interface te0/1
console(config-if)# udld port disable
console(config-if)# exit
```

MLD Snooping Commands

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`

Parameters

N/A

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
```

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, bridge multicast filtering must be enabled by the [bridge multicast filtering](#) command.

The user guidelines of the [bridge multicast mode](#) 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
```

ipv6 mld snooping querier

Use the `ipv6 mld snooping querier` Global Configuration mode command to enable globally the MLD Snooping querier. Use the `no` form of this command to disable the MLD Snooping querier globally.

Syntax

`ipv6 mld snooping querier`

no ipv6 mld snooping querier

Parameters

N/A

Default Configuration

Enabled

Command Mode

Global Configuration mode

User Guidelines

To run the MLD Snooping querier on a VLAN, you have enable it globally and on the VLAN.

Example

The following example disables the MLD Snooping querier globally:

```
console(config)# no ipv6 mld snooping querier
```

ipv6 mld snooping vlan querier

Use the `ipv6 mld snooping vlan querier` Global Configuration mode command to enable the Internet MLD Snooping querier on a specific VLAN. Use the `no` form of this command to disable the MLD Snooping querier on a VLAN interface.

Syntax

```
ipv6 mld snooping vlan vlan-id querier
```

```
no ipv6 mld snooping vlan vlan-id querier
```

Parameters

- *vlan-id*—Specifies the VLAN.

Default Configuration

Disabled

Command Mode

Global Configuration mode

User Guidelines

The MLD Snooping querier can be enabled on a VLAN only if MLD Snooping is enabled for that VLAN.

Example

The following example enables the MLD Snooping querier on VLAN 1:

```
console(config)# ipv6 mld snooping vlan 1 querier
```

ipv6 mld snooping vlan querier election

Use the `ipv6 mld snooping vlan querier election` Global Configuration mode command to enable MLD Querier election mechanism of an MLD Snooping querier on a specific VLAN. Use the `no` form of this command to disable Querier election mechanism.

Syntax

```
ipv6 mld snooping vlan vlan-id querier election
```

```
no ipv6 mld snooping vlan vlan-id querier election
```

Parameters

- *vlan-id*—Specifies the VLAN.

Default Configuration

Enabled

Command Mode

Global Configuration mode

User Guidelines

Use the `no` form of the `ipv6 mld snooping vlan querier election` command to disable MLD Querier election mechanism on a VLAN.

If the MLD Querier election mechanism is enabled, the MLD Snooping querier supports the standard MLD Querier election mechanism specified in RFC2710 and RFC3810.

If MLD Querier election mechanism is disabled, MLD Snooping Querier delays sending General Query messages for 60 seconds from the time it was enabled. During this time, if the switch did not receive an IGMP query from another Querier - it starts sending General Query messages. Once the switch acts as a Querier, it will stop sending General Query messages if it detects another Querier on the VLAN. In this case, the switch will resume sending General Query messages if it does hear another Querier for Query Passive interval that equals:

$$\langle \text{Robustness} \rangle * \langle \text{Query Interval} \rangle + 0.5 * \langle \text{Query Response Interval} \rangle.$$

See the [ipv6 mld robustness](#), [ipv6 mld query-interval](#), and [ipv6 mld query-max-response-time](#) commands for configurations of these parameters.

It is recommended to disable MLD Querier election mechanism if there is an IPMv6 Multicast router on the VLAN.

Example

The following example disables MLD Snooping Querier election on VLAN 1:

```
console(config)# no ipv6 mld snooping vlan 1 querier
election
```

ipv6 mld snooping vlan querier version

Use the `ipv6 mld 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

```
ipv6 mld snooping vlan vlan-id querier version {1 / 2}
no ipv6 mld snooping vlan vlan-id querier version
```

Parameters

- *vlan-id*—Specifies the VLAN.

- **querier version {1 / 2}**—Specifies the MLD version.

Default Configuration

MLDv1.

Command Mode

Global Configuration mode

Example

The following example sets the version of the MLD Snooping Querier VLAN 1 to 2:

```
console(config)# ipv6 mld snooping vlan 1 querier version  
2
```

ipv6 mld snooping vlan mrouter (Global)

Use the `ipv6 mld snooping vlan 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.
- `pim-dvmrp`—Learn Multicast router port by PIM, DVMRP and MLD messages.

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
```

ipv6 mld snooping vlan 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 and for a range of ports as shown in the example.

Example

```
console(config)# interface te0/1
console(config-if)# ipv6 mld snooping vlan 1 mrouter
interface te0/1-4
```

ipv6 mld snooping vlan forbidden mrouter

Use the `ipv6 mld snooping vlan forbidden mrouter` 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 of 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 to be defined as a Multicast router port (mrouter port) cannot be learned dynamically or assigned statically.

The **bridge multicast 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 te0/1
```

ipv6 mld snooping vlan static

Use the **ipv6 mld snooping vlan 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 FF12::3  
te0/1
```

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. When an MLD Leave Group message is received from a host, the system removes the host port from the table entry. After it relays the MLD queries from the Multicast router, it deletes entries periodically if it does not receive any MLD membership reports from the Multicast clients.

MLD snooping Immediate-Leave processing allows the switch to remove an interface that sends a leave message from the forwarding table without first sending out MAC-based general queries to the interface.

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
```

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

User EXEC mode

Default Configuration

Display information for all VLANs and addresses defined on them.


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 that 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 show ipv6 mld snooping groups.

```
console# show ipv6 mld snooping groups
```

VLAN	Group Address	Source Address	Include Ports	Exclude Ports
----	-----	-----	-----	-----
1	FF12::3	FE80::201:C9FF:FE40:8001	te0/1	
1	FF12::3	FE80::201:C9FF:FE40:8002	te0/2	
19	FF12::8	FE80::201:C9FF:FE40:8003	te0/4	
19	FF12::8	FE80::201:C9FF:FE40:8004	te0/1	te0/2

MLD Reporters that are forbidden statically:

VLAN	Group Address	Source Address	Ports
----	-----	-----	-----
1	FF12::3	FE80::201:C9FF:FE40:8001	te0/3
19	FF12::8	FE80::201:C9FF:FE40:8001	te0/4

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.

Default Configuration

Display information for all VLANs.

Command Mode

User 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 Querier is globally enabled
VLAN 1000
  MLD Snooping is enabled
  MLD snooping last immediate leave: enable
  Automatic learning of multicast router ports is enabled
  MLD Snooping Querier is enabled
  MLD Snooping Querier operation state: is running
  MLD Snooping Querier version: 2
  MLD Snooping Querier election is enabled
  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 500 msec
  Groups that are in MLD version 1 compatibility mode:
    FF12::3, FF12::8
```

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.

Default Configuration

Display information for all VLANs.

Command Mode

User 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	Dynamic	Static	Forbidden
----	-----	-----	-----
1000	te0/1	te0/2	te0/3-4

MLD Commands

ipv6 mld last-member-query-count

To configure the Multicast Listener Discovery (MLD) last member query counter, use the `ipv6 mld last-member-query-count` command in interface configuration mode. To restore the default value, use the `no` form of this command.

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 a message indicating a leave. (Range: 1–7)

Default Configuration

A value of MLD Robustness variable.

Command Mode

Interface configuration

User Guidelines

Use the `ipv6 mld robustness` command to change the MLD last member query counter.

Example

The following example changes a value of the MLD last member query counter to 3:

```
interface vlan 1
  ipv6 mld last-member-query-count 3
```

exit

ipv6 mld last-member-query-interval

To configure the Multicast Listener Discovery (MLD) last member query interval, use the **ipv6 mld last-member-query-interval** command in interface configuration mode. To restore the default MLD query interval, use the **no** form of this command.

Syntax

```
ipv6 mld last-member-query-interval milliseconds  
no ipv6 mld last-member-query-interval
```

Parameters

- *milliseconds*—Interval, in milliseconds, at which MLD group-specific host query messages are sent on the interface. (Range: 100–25500).

Default Configuration

The default MLD last member query interval is 1000 milliseconds.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

Use the **ipv6 mld last-member-query-interval** command to configure the MLD last member query interval on an interface.

Example

The following example shows how to increase the MLD last member query interval to 1500 milliseconds:

```
console(config)# interface vlan 100  
console(config-if)# ipv6 mld last-member-query-interval  
1500  
console(config-if)# exit
```

ipv6 mld query-interval

To configure the frequency at which the switch sends Multicast Listener Discovery (MLD) host-query messages, use the **ipv6 mld query-interval** command in interface configuration mode. To return to the default frequency, use the **no** form of this command.

Syntax

ipv6 mld query-interval *seconds*

no ipv6 mld query-interval

Parameters

- *seconds*—Frequency, in seconds, at which the switch sends MLD query messages from the interface. The range is from 1 to 31744.

Default Configuration

The default MLD query interval is 125 seconds.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

Use the **ipv6 mld query-interval** command to configure the frequency at which the MLD querier sends MLD host-query messages from an interface. The MLD querier sends query-host messages to discover which multicast groups have members on the attached networks of the router.

The query interval must be bigger than the maximum query response time.

Example

The following example shows how to increase the frequency at which the MLD querier sends MLD host-query messages to 180 seconds:

```
console(config)# interface vlan 100
console(config-if)# ipv6 mld query-interval 180
console(config-if)# exit
```

ipv6 mld query-max-response-time

To configure the maximum response time advertised in Multicast Listener Discovery (MLD) queries, use the **ipv6 mld query-max-response-time** command in interface configuration mode. To restore the default value, use the **no** form of this command.

Syntax

ipv6 mld query-max-response-time *seconds*

no ipv6 mld query-max-response-time

Parameters

- *seconds*—Maximum response time, in seconds, advertised in MLD queries. (Range: 0–31744)

Default Configuration

10 seconds.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

This command controls the period during which the responder can respond to an MLD query message before the router deletes the group.

This command controls how much time the hosts have to answer an MLD query message before the router deletes their group. Configuring a value of fewer than 10 seconds enables the router to prune groups faster.

The maximum query response time must be less than the query interval.



NOTE: If the hosts do not respond fast enough, they might be pruned inadvertently. Therefore, the hosts must know to respond faster than 10 seconds (or the value you configure).

Example

The following example configures a maximum response time of 8 seconds:

```
console(config)# interface vlan 100
console(config-if)# ipv6 mld query-max-response-time 8
console(config-if)# exit
```

ipv6 mld robustness

To configure the Multicast Listener Discovery (MLD) robustness variable, use the **ipv6 mld robustness** command in interface configuration mode. To restore the default value, use the **no** form of this command.

Syntax

ipv6 mld robustness *count*

no ipv6 mld robustness

Parameters

- *count*—The number of expected packet loss on a link. Parameter range. (Range: 1–7).

Default Configuration

The default value is 2.

Command Mode

Interface (VLAN) Configuration mode

User Guidelines

Use the **ipv6 mld robustness** command to change the MLD robustness variable.

Example

The following example changes a value of the MLD robustness variable to 3:

```
console(config)# interface vlan 1
```

```
console(config-if)# ipv6 mld robustness 3  
console(config-if)# exit
```

Port Features

Overview

The network ports of the switch are divided into internal and external ports. The internal ports are connected to blade servers cards through the mid-plan of the chassis.

Supported Features on Ports

Feature	VRTX 10G		
	External 10G Ports	Internal 10G Ports	External 1G Ports
Administrative Port Status (shutdown)	Supported	Supported	Supported
Port Speed	1000Mbit/sec or 10Gbit/sec	N/A	10/100/1000 Mbits/sec
Duplex Mode	Full Only	N/A	Full/Half
Auto Negotiation	AP 1G/10G 802.3ap	AP 1G/10G 802.3ap	Enable /Disable
Flow Control	Enable/Disable	AP 1G/10G 802.3ap	Enable /Disable
MDIX	N/A	N/A	Auto/MDI/MDIX
Green Ethernet	N/A	N/A	Supported
Cable Diagnostic	N/A	N/A	Supported

In the CLI, an acknowledgment message is displayed to indicate when the features above are not supported, as shown below:

Example 1: This example shows the error message received when a non-supported feature is requested.

```
console(config)# interface gil/1
console(config-if)# green-ethernet short-reach
The feature is not supported on the internal port gil/1
console(config-if)#
```

Example 2: This example shows that running the command that disables short-reach on an internal port is ignored (there is no reaction by the system).

```
console(config)# interface gil/1
console(config-if)# no green-ethernet short-reach
console(config-if)#
```

Example 3: This example shows the error message received when a non-supported feature is requested on range of ports:

```
console(config)# interface gil/1-4
console(config-if-range)# green-ethernet short-reach
The feature is not supported on the internal port gil/1.
The feature is not supported on th internal port gil/2.
The feature is not supported on the internal port gil/3.
The feature is not supported on the internal port gil/4
```