

■ connecting your business



CLI Reference

GS-2300 series

Contents

1 Operation of CLI Management.....	14
2 AAA Commands.....	19
2.1 <i>acc-radius</i>	19
2.2 <i>accounting</i>	20
2.3 <i>authorization</i>	21
2.4 <i>deadtime</i>	21
2.5 <i>fallback-author</i>	22
2.6 <i>radius</i>	22
2.7 <i>show</i>	23
2.8 <i>tacacs+</i>	25
2.9 <i>timeout</i>	26
3 Access Commands.....	27
3.1 <i>add</i>	27
3.2 <i>clear</i>	28
3.3 <i>delete</i>	28
3.4 <i>mode</i>	29
3.5 <i>show</i>	29
4 Account Commands.....	31
4.1 <i>add</i>	31
4.2 <i>delete</i>	32
4.3 <i>show</i>	32
5 ACL Commands.....	33
5.1 <i>ace</i>	33
5.1.1 <i>action</i>	35
5.1.2 <i>arp-opcode</i>	35
5.1.3 <i>arp-flags</i>	36
5.1.4 <i>dip</i>	37
5.1.5 <i>dmac</i>	37
5.1.6 <i>dport</i>	38
5.1.7 <i>end</i>	38
5.1.8 <i>etype</i>	39
5.1.9 <i>icmp-code</i>	39
5.1.10 <i>icmp-type</i>	40
5.1.11 <i>ip-flags</i>	40
5.1.12 <i>ip-protocol</i>	41
5.1.13 <i>logging</i>	41
5.1.14 <i>mirror</i>	42
5.1.15 <i>rate</i>	42
5.1.16 <i>show</i>	43

5.1.17 <i>shutdown</i>	43
5.1.18 <i>sip</i>	43
5.1.19 <i>smac</i>	44
5.1.20 <i>sport</i>	44
5.1.21 <i>tcp-flags</i>	45
5.1.22 <i>vlan</i>	46
5.2 <i>action</i>	46
5.3 <i>clear</i>	47
5.4 <i>delete</i>	48
5.5 <i>logging</i>	48
5.6 <i>mirror</i>	49
5.7 <i>move</i>	49
5.8 <i>policy</i>	50
5.9 <i>port-rate</i>	50
5.10 <i>rate-limiter</i>	51
5.11 <i>show</i>	51
5.12 <i>shutdown</i>	52
5.13 <i>state</i>	53
6 Aggregation Commands.....	54
6.1 <i>delete</i>	54
6.2 <i>group</i>	54
6.3 <i>mode</i>	55
6.4 <i>show</i>	56
7 Arp-inspection Commands.....	57
7.1 <i>add</i>	57
7.2 <i>delete</i>	57
7.3 <i>mode</i>	58
7.4 <i>port-mode</i>	58
7.5 <i>show</i>	59
7.6 <i>translate</i>	60
8 Auth Method Commands.....	61
8.1 <i>fallback</i>	61
8.2 <i>method</i>	61
8.3 <i>show</i>	62
9 Config-file Commands.....	63
9.1 <i>export</i>	63
9.2 <i>import</i>	63
10 DHCP Relay Commands.....	65
10.1 <i>clear</i>	65
10.2 <i>mode</i>	65
10.3 <i>relay-option</i>	66
10.4 <i>server</i>	66
10.5 <i>show</i>	67

11 DHCP Snooping Commands.....	69
11.1 <i>clear</i>	69
11.2 <i>mode</i>	69
11.3 <i>port-mode</i>	70
11.4 <i>show</i>	71
12 Diagnostic Commands.....	73
12.1 <i>ping</i>	73
12.2 <i>ping6</i>	74
12.3 <i>veriphy</i>	74
13 Easyport Commands.....	76
13.1 <i>ip-cam</i>	76
13.1.1 <i>access-vlan</i>	77
13.1.2 <i>admin-edge</i>	77
13.1.3 <i>bpdu-guard</i>	77
13.1.4 <i>end</i>	78
13.1.5 <i>psec-action</i>	78
13.1.6 <i>psec-limit</i>	79
13.1.7 <i>psec-mode</i>	79
13.1.8 <i>show</i>	79
13.1.9 <i>traffic-class</i>	80
13.1.10 <i>vlan-mode</i>	80
13.2 <i>ip-phone</i>	81
13.2.1 <i>access-vlan</i>	81
13.2.2 <i>admin-edge</i>	82
13.2.3 <i>bpdu-guard</i>	82
13.2.4 <i>end</i>	83
13.2.5 <i>psec-action</i>	83
13.2.6 <i>psec-limit</i>	83
13.2.7 <i>psec-mode</i>	84
13.2.8 <i>show</i>	84
13.2.9 <i>traffic-class</i>	84
13.2.10 <i>vlan-mode</i>	85
13.3 <i>wifi-ap</i>	85
13.3.1 <i>access-vlan</i>	86
13.3.2 <i>admin-edge</i>	86
13.3.3 <i>bpdu-guard</i>	87
13.3.4 <i>end</i>	87
13.3.5 <i>psec-action</i>	87
13.3.6 <i>psec-limit</i>	88
13.3.7 <i>psec-mode</i>	88
13.3.8 <i>show</i>	89
13.3.9 <i>traffic-class</i>	89
13.3.10 <i>vlan-mode</i>	89

14 EEE Commands.....	91
14.1 <i>mode</i>	91
14.2 <i>show</i>	92
14.3 <i>urgent-queue</i>	92
15 Event Commands.....	94
15.1 <i>group</i>	94
15.2 <i>show</i>	95
16 Fdb Commands.....	97
16.1 <i>age-time</i>	97
16.2 <i>delete</i>	98
16.3 <i>flush</i>	98
16.4 <i>learning</i>	99
16.5 <i>static-mac</i>	99
16.6 <i>show</i>	100
17 Firmware Commands.....	102
17.1 <i>show</i>	102
17.2 <i>swap</i>	102
17.3 <i>upgrade</i>	103
18 GARP Commands.....	105
18.1 <i>applicant</i>	105
18.2 <i>join-time</i>	106
18.3 <i>leave-all</i>	106
18.4 <i>leave-time</i>	107
18.5 <i>show</i>	107
19 GVRP Commands	109
19.1 <i>clear</i>	109
19.2 <i>control</i>	109
19.3 <i>mode</i>	110
19.4 <i>rrole</i>	110
19.5 <i>show</i>	111
20 HTTPs Commands.....	113
20.1 <i>mode</i>	113
20.2 <i>redirect</i>	113
20.3 <i>show</i>	114
21 IGMP Commands.....	115
21.1 <i>compatibility</i>	115
21.2 <i>delete</i>	116
21.3 <i>fast-leave</i>	117
21.4 <i>filtering</i>	117
21.5 <i>flooding</i>	118
21.6 <i>lmqi</i>	118
21.7 <i>proxy</i>	119
21.8 <i>qi</i>	120

21.9	<i>qri</i>	120
21.10	<i>querier</i>	121
21.11	<i>router</i>	121
21.12	<i>rv</i>	122
21.13	<i>show</i>	122
21.14	<i>snooping</i>	123
21.15	<i>ssm-range</i>	123
21.16	<i>state</i>	124
21.17	<i>throttling</i>	124
21.18	<i>uri</i>	125
22	IP Commands.....	126
22.1	<i>dhcp</i>	126
22.2	<i>dns-proxy</i>	127
22.3	<i>mgmt-vlan</i>	127
22.4	<i>name-server</i>	128
22.5	<i>setup</i>	128
22.6	<i>show</i>	129
23	IP-Source-Guard Commands.....	130
23.1	<i>add</i>	130
23.2	<i>delete</i>	131
23.3	<i>limit</i>	131
23.4	<i>mode</i>	132
23.5	<i>port-mode</i>	133
23.6	<i>show</i>	134
23.7	<i>translate</i>	135
24	IPv6 Commands.....	136
24.1	<i>autoconfig</i>	136
24.2	<i>setup</i>	136
24.3	<i>show</i>	137
25	LACP Commands.....	139
25.1	<i>clear</i>	139
25.2	<i>key</i>	139
25.3	<i>mode</i>	140
25.4	<i>role</i>	141
25.5	<i>show</i>	141
26	LLDP Commands	143
26.1	<i>cdp-aware</i>	143
26.2	<i>clear</i>	144
26.3	<i>delay</i>	144
26.4	<i>hold</i>	145
26.5	<i>interval</i>	146
26.6	<i>mode</i>	146
26.7	<i>option-tiv</i>	147

26.8	<i>reinit</i>	148
26.9	<i>show</i>	149
27	LLDP Media Commands.....	151
27.1	<i>civic</i>	151
27.2	<i>coordinate</i>	153
27.3	<i>delete</i>	155
27.4	<i>ecs</i>	155
27.5	<i>fast</i>	156
27.6	<i>policy</i>	157
27.7	<i>port-policy</i>	158
27.8	<i>show</i>	159
28	Loop protection Commands.....	161
28.1	<i>interval</i>	161
28.2	<i>mode</i>	161
28.3	<i>port-action</i>	162
28.4	<i>port-mode</i>	163
28.5	<i>port-transmit</i>	164
28.6	<i>show</i>	164
28.7	<i>shutdown</i>	165
29	Port Mirroring Commands.....	166
29.1	<i>analyzer-port</i>	166
29.2	<i>port-mode</i>	166
29.3	<i>show</i>	167
30	MLD Commands.....	169
30.1	<i>compatibility</i>	169
30.2	<i>delete</i>	170
30.3	<i>fast-leave</i>	170
30.4	<i>filtering</i>	171
30.5	<i>flooding</i>	172
30.6	<i>lmqi</i>	173
30.7	<i>proxy</i>	173
30.8	<i>qi</i>	173
30.9	<i>qri</i>	174
30.10	<i>querier</i>	175
30.11	<i>router</i>	176
30.12	<i>rv</i>	176
30.13	<i>show</i>	177
30.14	<i>snooping</i>	178
30.15	<i>ssm-range</i>	178
30.16	<i>state</i>	179
30.17	<i>throtting</i>	180
30.18	<i>uri</i>	180
31	MVR Commands.....	182

31.1 <i>allow</i>	182
31.2 <i>delete</i>	182
31.3 <i>immediate-leave</i>	183
31.4 <i>mode</i>	184
31.5 <i>port-mode</i>	184
31.6 <i>port-type</i>	185
31.7 <i>show</i>	185
32 NAS Commands.....	187
32.1 <i>agetime</i>	187
32.2 <i>clear</i>	188
32.3 <i>eapol-timeout</i>	188
32.4 <i>guest-vlan</i>	189
32.5 <i>hold-time</i>	190
32.6 <i>mode</i>	190
32.7 <i>port-guest-vlan</i>	191
32.8 <i>port-radius-qos</i>	192
32.9 <i>port-radius-vlan</i>	192
32.10 <i>port-state</i>	193
32.11 <i>radius-qos</i>	194
32.12 <i>radius-vlan</i>	194
32.13 <i>reauth-period</i>	195
32.14 <i>reauthentication</i>	196
32.15 <i>restart</i>	196
32.16 <i>show</i>	197
33 PoE Commands.....	199
33.1 <i>delay-mode</i>	199
33.2 <i>delay-time</i>	200
33.3 <i>detection</i>	200
33.4 <i>failure-action</i>	201
33.5 <i>hour</i>	201
33.6 <i>interval-time</i>	202
33.7 <i>max-power</i>	202
33.8 <i>mode</i>	203
33.9 <i>ping-check</i>	203
33.10 <i>ping-ip-addr</i>	204
33.11 <i>ping-retry-time</i>	204
33.12 <i>priority</i>	205
33.13 <i>reboot-time</i>	205
33.14 <i>reset-port</i>	206
33.15 <i>retry-time</i>	206
33.16 <i>schedule-mode</i>	207
33.17 <i>select-all</i>	207
33.18 <i>show</i>	208
34 Port configuration Commands	209

34.1	<i>clear</i>	209
34.2	<i>description</i>	210
34.3	<i>excessive-collision</i>	210
34.4	<i>flow-control</i>	211
34.5	<i>max-frame</i>	212
34.6	<i>port-state</i>	213
34.7	<i>power-saving</i>	213
34.8	<i>show</i>	214
34.9	<i>speed-duplex</i>	216
35	Port security Commands	217
35.1	<i>action</i>	217
35.2	<i>aging</i>	218
35.3	<i>limit</i>	218
35.4	<i>mode</i>	219
35.5	<i>port-mode</i>	219
35.6	<i>reopen</i>	220
35.7	<i>show</i>	221
36	Privilege level Commands.....	223
36.1	<i>group</i>	223
36.2	<i>show</i>	224
37	Private VLAN Commands.....	226
37.1	<i>delete</i>	226
37.2	<i>port-isolate</i>	226
37.3	<i>private-vlan</i>	227
37.4	<i>show</i>	227
38	QoS Commands.....	229
38.1	<i>delete</i>	230
38.2	<i>dscp-classification</i>	230
38.3	<i>dscp-map</i>	231
38.4	<i>dscp-remap</i>	232
38.5	<i>dscp-translation</i>	232
38.6	<i>dscp-trust</i>	233
38.7	<i>port-classify</i>	234
38.8	<i>port-dscp</i>	237
38.9	<i>port-policer</i>	239
38.10	<i>port-scheduler</i>	240
38.11	<i>port-shaper</i>	241
38.12	<i>qce</i>	242
38.12.1	<i>class</i>	244
38.12.2	<i>classified-dscp</i>	244
38.12.3	<i>dei</i>	245
38.12.4	<i>dmac</i>	246
38.12.5	<i>dp</i>	246

38.12.6 <i>end</i>	247
38.12.7 <i>pcp</i>	247
38.12.8 <i>show</i>	248
38.12.9 <i>smac</i>	249
38.12.10 <i>tag</i>	250
38.12.11 <i>vid</i>	250
38.13 <i>queue-shaper</i>	251
38.14 <i>show</i>	252
38.15 <i>storm</i>	256
38.16 <i>tag-remarking</i>	257
39 Reboot Commands.....	259
39.1 <i>reboot</i>	259
40 SFlow Commands.....	260
40.1 <i>collector</i>	260
40.2 <i>sampler</i>	261
40.3 <i>show</i>	262
41 Single IP Commands.....	263
41.1 <i>connect</i>	263
41.2 <i>group-name</i>	263
41.3 <i>mode</i>	264
41.4 <i>show</i>	264
42 SMTP Commands.....	266
42.1 <i>delete</i>	266
42.2 <i>level</i>	267
42.3 <i>mail-address</i>	268
42.4 <i>return-path</i>	268
42.5 <i>sender</i>	269
42.6 <i>server</i>	269
42.7 <i>show</i>	270
42.8 <i>username</i>	270
43 SNMP Commands.....	272
43.1 <i>access</i>	272
43.2 <i>community</i>	273
43.3 <i>delete</i>	274
43.4 <i>engine-id</i>	275
43.5 <i>getcommunity</i>	275
43.6 <i>group</i>	276
43.7 <i>mode</i>	277
43.8 <i>setcommunity</i>	277
43.9 <i>show</i>	278
43.10 <i>trap</i>	279
43.11 <i>user</i>	280
43.12 <i>view</i>	281

44 SSH Commands.....	283
44.1 <i>mode</i>	283
44.2 <i>show</i>	283
45 STP Commands.....	284
45.1 <i>cname</i>	285
45.2 <i>fwdDelay</i>	285
45.3 <i>maxage</i>	286
45.4 <i>maxhops</i>	286
45.5 <i>statistics</i>	287
45.6 <i>txhold</i>	287
45.7 <i>version</i>	288
45.8 <i>bpdufilter</i>	288
45.9 <i>bpduguard</i>	289
45.10 <i>migrate-check</i>	289
45.11 <i>msti-vlan</i>	290
45.12 <i>p-autoEdge</i>	291
45.13 <i>p-bpduguard</i>	292
45.14 <i>p-cost</i>	292
45.15 <i>p-edge</i>	293
45.16 <i>p-mode</i>	294
45.17 <i>p-p2p</i>	295
45.18 <i>p-priority</i>	296
45.19 <i>priority</i>	297
45.20 <i>r-role</i>	297
45.21 <i>r-tcn</i>	298
45.22 <i>recovery</i>	299
45.23 <i>show</i>	299
46 Syslog Commands.....	301
46.1 <i>clear</i>	301
46.2 <i>level</i>	301
46.3 <i>mode</i>	302
46.4 <i>server</i>	303
46.5 <i>show</i>	303
47 System Commands.....	305
47.1 <i>contact</i>	305
47.2 <i>location</i>	305
47.3 <i>name</i>	306
47.4 <i>show</i>	306
48 Thermal Commands.....	308
48.1 <i>port-priority</i>	308
48.2 <i>priority-temp</i>	309
48.3 <i>show</i>	309
49 Time Commands.....	311

49.1	<i>clock-source</i>	311
49.2	<i>daylight</i>	311
49.3	<i>delete</i>	313
49.4	<i>manual</i>	314
49.5	<i>ntp</i>	314
49.6	<i>show</i>	315
49.7	<i>time-zone</i>	315
50	UPnP Commands.....	317
50.1	<i>duration</i>	317
50.2	<i>mode</i>	317
50.3	<i>show</i>	318
50.4	<i>ttl</i>	318
51	VCL Commands.....	319
51.1	<i>delete</i>	319
51.2	<i>mac-vlan</i>	320
51.3	<i>protocol-vlan</i>	321
51.4	<i>show</i>	322
52	VLAN Commands.....	324
52.1	<i>delete</i>	324
52.2	<i>egress-rule</i>	325
52.3	<i>forbidden</i>	325
52.4	<i>frame-type</i>	326
52.5	<i>ingress-filtering</i>	327
52.6	<i>port-type</i>	327
52.7	<i>pvid</i>	328
52.8	<i>show</i>	329
52.9	<i>tag-group</i>	330
52.10	<i>tpid</i>	331
53	Voice VLAN Commands.....	332
53.1	<i>config</i>	332
53.2	<i>delete</i>	333
53.3	<i>discovery</i>	333
53.4	<i>oui</i>	334
53.5	<i>port-mode</i>	335
53.6	<i>security</i>	335
53.7	<i>show</i>	336
54	Global Commands.....	338
54.1	<i>auto-logout</i>	338
54.2	<i>exit</i>	338
54.3	<i>help</i>	339
54.4	<i>history</i>	339
54.5	<i>logout</i>	340
54.6	<i>restore</i>	340

54.7 <i>save</i>	341
------------------------	-----

1 Operation of CLI Management

Initial Configuration

This chapter instructs you how to configure and manage the switch through the CLI interface. With this facility, you can easily access and monitor through console port of the switch all the status of the switch, including MIBs status, each port activity, Spanning tree status, port aggregation status, multicast traffic, VLAN and priority status, even illegal access record and so on.

The serial port's configuration requirements are as follows:

- Default Baud rate: 115,200 bps
- Character Size: 8 Characters
- Parity: None
- Stop bit: One
- Data bits: 8
- Flow control: none

About Null Console Cable identity

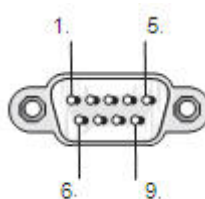


Figure 1: Serial Port Pin Definition (DB-9 DTE)

The DB-9 cable is used for connecting a terminal or terminal emulator to the Managed Switch's RS-232 port to access the command-line interface. The table below shows the pin assignments for the DB-9 cable.

Function	Mnemonic	Pin
Carrier	CD	1
Receive Data	RXD	2
Transmit Data	TXD	3
Data Terminal Ready	DTR	4
Signal Ground	GND	5
Data Set Ready	DSR	6
Request To Send	RTS	7
Clear To Send	CTS	8

 The model GS-2326(P) is equipped with a serial RJ45 port instead.

Connecting to the console port

The serial port on the switch's front panel is used to connect to the switch for out-of-band console configuration.

The command-line-driven configuration program can be accessed from a terminal or a PC running a terminal emulation program. The pin assignments used to connect to the serial port are provided in the following table

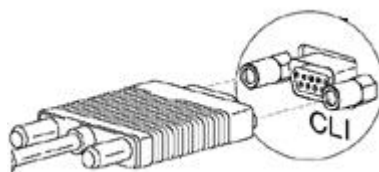


Figure 2: Plug in the Console Port (DB-9 DTE)

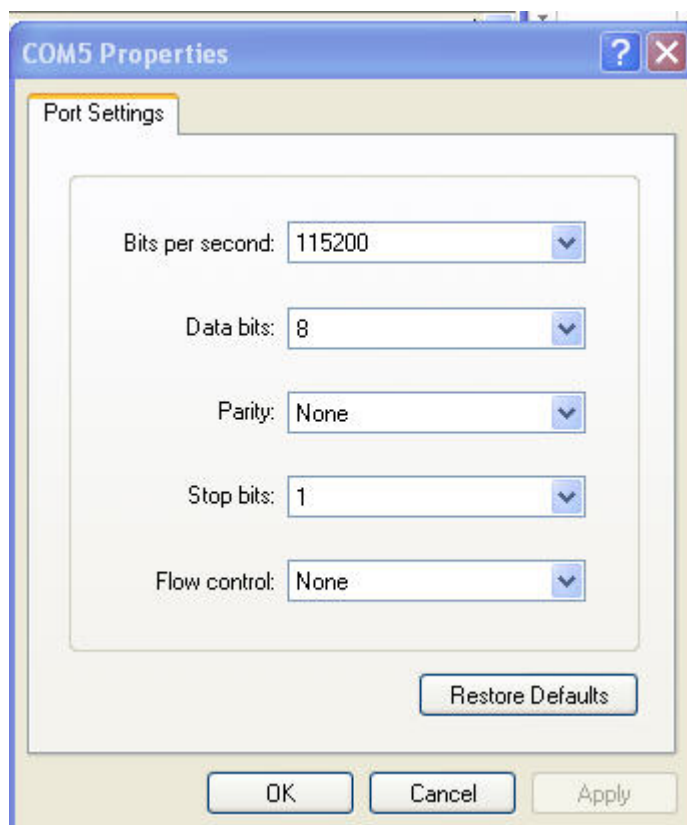


Figure 3: Console configure

After the switch has been finished configuration the it interface, you can access It via Console port. For instance, it will show the following screen and ask you inputting username and password in order to login and access authentication.

The default username and password is `admin`. For the first time to use, please enter the default username and password, and then click the Enter button. The login process now is completed. In this login menu, you have to input the complete

username and password respectively, the switch will not give you a shortcut to username automatically. This looks inconvenient, but safer.

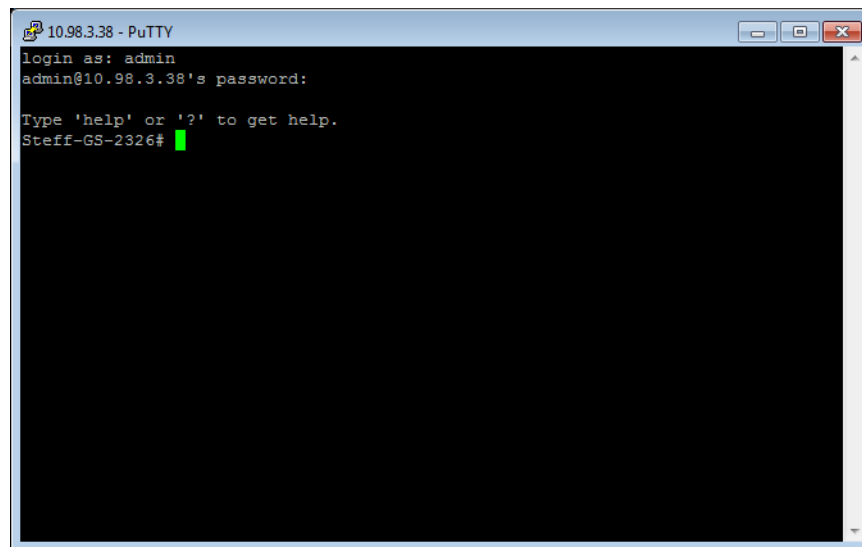


Figure 4: Console configure



You can type `?` or `help` to get the switch help includes syntax or all function explaining. The screen shot as below figure displayed.

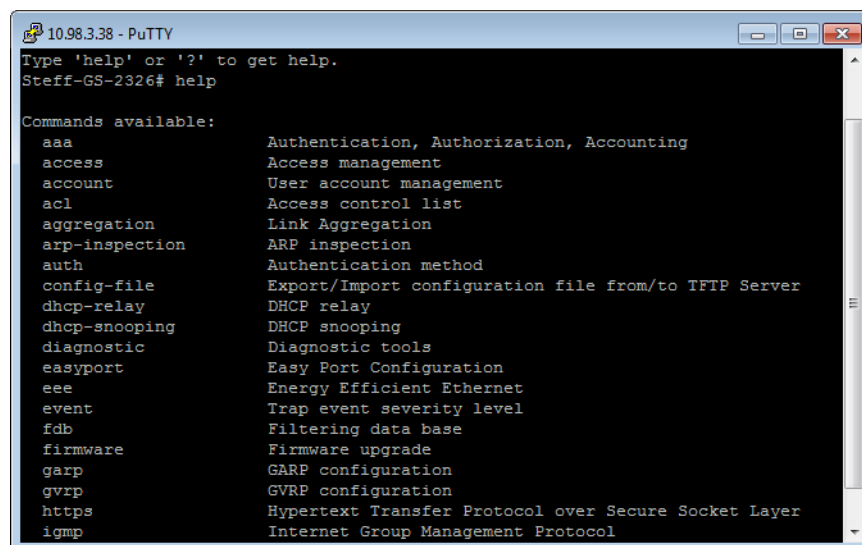


Figure 5: Console configure

Connecting via SSH using PuTTY

The connection to the switch can also be established via SSH connection using tools like PuTTY:

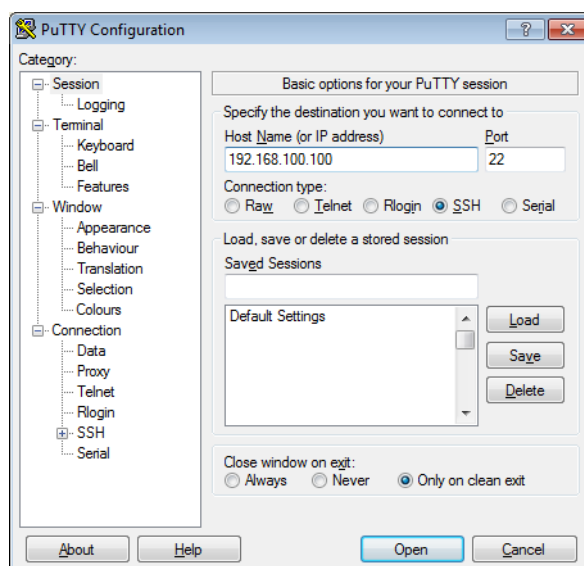


Figure 6: PuTTY configure

Navigating through CLI structure

After you have established a connection to the CLI, you can navigate through the CLI structure just by entering the name of the required command section. For example, if you want to change AAA parameters, just enter `aaa` at the command line. The prompt will show the current menu in round brackets. Here you can enter the required commands, e.g. `show config`:

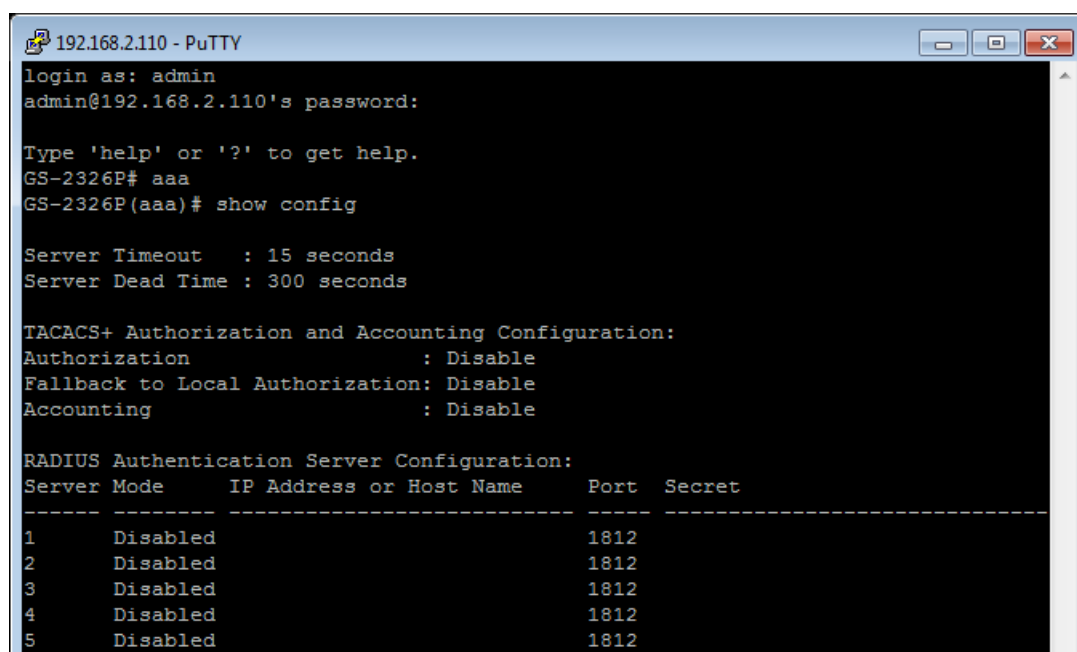


Figure 7: PuTTY configure

For some of the commands you will find a 2-level structure of commands, e.g. ACL/ace or QoS/qce. In those cases you will enter the first level command including all required or optional parameters. After the 1st level command has been

entered, the prompt will show the menu and the first level command in round brackets. Here you can enter the 2nd level commands including all required parameters. Note that you might want to execute more than one 2nd level commands. After all 2nd level commands have been entered, you will finish this sequence using the **end** command, which will bring you back to the first level menu.

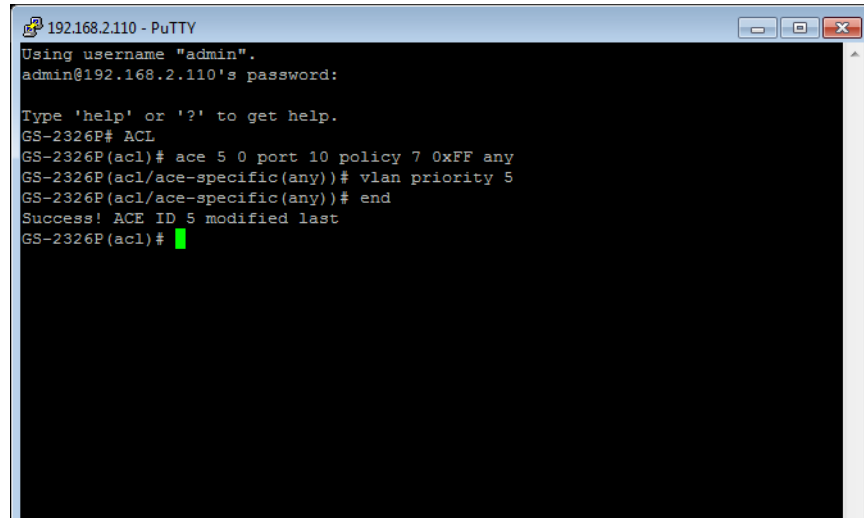
A screenshot of a PuTTY terminal window titled "192.168.2.110 - PuTTY". The terminal shows a login sequence where the username "admin" is entered. The prompt then changes to "admin@192.168.2.110's password:". After the password is entered, the prompt changes to "Type 'help' or '?' to get help." followed by "GS-2326P# ACL". The user then enters a series of commands: "ace 5 0 port 10 policy 7 0xFF any", "vlan priority 5", and "end". The terminal displays "Success! ACE ID 5 modified last" and returns to the "GS-2326P# " prompt with a green cursor.

Figure 8: PuTTY configure

2 AAA Commands

This section shows you how to use an AAA (Authentication, Authorization, Accounting) server to provide access control to your network. The AAA server can be a TACACS+ or RADIUS server to create and manage objects that contain settings for using AAA servers.

Command	Function
acc-radius	Configure RADIUS accounting Server
accounting	Configure Accounting mode
authorization	Configure Authorization mode
deadtime	Configure server dead time
fallback-author	Configure Authorization mode
radius	Configure RADIUS authentication server
show	Show AAA information
tacacs+	Configure TACACS+ authentication server
timeout	Configure server response timeout

2.1 *acc-radius*

Using this command you can configure the RADIUS accounting server parameters.

Syntax:

```
acc-radius <index> enable [<ip-hostname>] [<0-65535>] [<Line>]  
acc-radius <index> disable
```

Parameter:

<index>

The index of the RADIUS accounting server.

Possible values: 1 to 5.

enable|disable

Enables or disables the RADIUS accounting for the given index.

<ip-hostname>

The IP address or hostname of the RADIUS accounting server.

<0-65535>

The UDP port used for connection to RADIUS accounting server.

Special values: 0 is equivalent to default port (1813).

<LINE>

Shared secret for accessing the external server.

Possible values: maximum 29 characters.

Example:

```

Switch(aaa)# acc-radius 1 enable 192.168.2.22 65535 radius
Switch(aaa)# show config

Server Timeout      : 15 seconds
Server Dead Time    : 300 seconds
TACACS+ Authorization and Accounting Configuration:
Authorization       : Disable
Fallback to Local Authorization: Disable
Accounting          : Disable

RADIUS Authentication Server Configuration:
Server Mode      IP Address or Host Name      Port  Secret
-----
RADIUS Authentication Server Configuration:
Server Mode      IP Address or Host Name      Port  Secret
-----
1      Disabled      1812
2      Disabled      1812
3      Disabled      1812
4      Disabled      1812
5      Disabled      1812

RADIUS Accounting Server Configuration:
Server Mode      IP Address or Host Name      Port  Secret
-----
1      Enabled  192.168.2.22      65535 radius
2      Disabled      1813
3      Disabled      1813
4      Disabled      1813
5      Disabled      1813

TACACS+ Authentication Server Configuration:
Server Mode      IP Address or Host Name      Port  Secret
-----
1      Disabled      49
2      Disabled      49
3      Disabled      49
4      Disabled      49
5      Disabled      49
Switch(aaa)#

```

2.2 accounting

Using this command you can enable or disable the global RADIUS accounting operation mode.



If no RADIUS accounting server is configured or reachable when enabling the global RADIUS accounting operation mode, the device will show "Server disconnect".

Syntax:

```
accounting enable|disable
```

Parameter:**enable**

Globally enable accounting operation mode.

disable

Globally disable accounting operation mode.

Example:

```
Switch(aaa)# accounting enable
Server disconnect!
Switch(aaa)# accounting disable
Switch(aaa)#
```

2.3 authorization

Using this command you can enable or disable the TACACS+/RADIUS authorization mode.



Never enable remote authentication if no authentication server is configured or reachable AND *fallback to local authentication* is disabled. If you enable remote authentication with no server available and the fallback setting disabled, the device will decline any further action with the authentication error "Server disconnect!". In this state, remote administration is prohibited and the device can only be re-accessed by performing a hard reset.

Syntax:

```
authorization enable|disable
```

Parameter:**enable**

Globally enable TACACS+/RADIUS authorization operation mode.

disable

Globally disable TACACS+/RADIUS authorization operation mode.

Example:

```
Switch(aaa)# authorization enable
Switch(aaa)#
```

2.4 deadtime

Using this command you can configure the RADIUS server deadtime.

The deadtime is the period during which the switch will not send new requests to a server that has failed to respond to a previous request. This will stop the switch from continually trying to contact a server that it has already determined as dead.

Syntax:

```
deadtime <0-3600>
```

Parameter:**<0-3600>**

Time that a server is considered dead if it doesn't answer a request.

Possible values: 0 to 3600 seconds.

Special values: 0 disables this feature.

Example:

```
Switch(aaa)# deadtime 3600
Server disconnect!
Switch(aaa)#
```

2.5 *fallback-author*

Using this command you can configure the fallback function of RADIUS authorization in case the remote authorization fails.

Syntax:

```
fallback-author enable|disable
```

Parameter:**enable**

Enables fallback function in case remote authorization fails.

disable

Disables fallback function.

Example:

```
Switch(aaa)# fallback-author enable
Server disconnect!
```

2.6 *radius*

Using this command you can configure the RADIUS authentication server parameters.



If no RADIUS authentication server is configured or reachable, the device will show "Server disconnect".

Syntax:

```
radius <index> enable [<ip-hostname>] [<0-65535>] [<Line>]  
radius <index> disable
```

Parameter:**<index>**

The index of the RADIUS authentication server.

Possible values: 1 to 5.

enable|disable

Enables or disables the RADIUS authentication for the given index.

<ip-hostname>

The IP address or hostname of the RADIUS authentication server.

<0-65535>

The UDP port used for connection to RADIUS authentication server.

Special values: 0 is equivalent to default port (1813).

<LINE>

Shared secret for accessing the external server.

Possible values: maximum 29 characters.

Example:

```
Switch(aaa)# radius 1 enable 192.168.2.22 0 radius  
Server disconnect!
```

2.7 show

Using this command you can display the current RADIUS AAA information.

Syntax:

```
show config  
show statistics <1-5>
```

Parameter:**config**

Shows AAA configuration.

statistics

Shows RADIUS statistics.

<1-5>

The index of the RADIUS server you want to show statistics for.

Example:

```
Switch(aaa)# show config

Server Timeout      : 15 seconds
Server Dead Time    : 300 seconds

TACACS+ Authorization and Accounting Configuration:
Authorization        : Disable
Fallback to Local Authorization: Disable
Accounting           : Disable

RADIUS Authentication Server Configuration:
Server Mode          IP Address or Host Name      Port  Secret
-----
1      Disabled      1812
2      Disabled      1812
3      Disabled      1812
4      Disabled      1812
5      Disabled      1812

RADIUS Accounting Server Configuration:
Server Mode          IP Address or Host Name      Port  Secret
-----
1      Disabled      1813
2      Disabled      1813
3      Disabled      1813
4      Disabled      1813
5      Disabled      1813

TACACS+ Authentication Server Configuration:
Server Mode          IP Address or Host Name      Port  Secret
-----
1      Disabled      49
2      Disabled      49
3      Disabled      49
4      Disabled      49
5      Disabled      49
Switch(aaa)#

Switch(aaa)# show statistics 1

Server #1 (0.0.0.0:1812) RADIUS Authentication Statistics:
Rx Access Accepts          0 Tx Access Requests
0
Rx Access Rejects          0 Tx Access Retransmissions
0
Rx Access Challenges        0 Tx Pending Requests
0
Rx Malformed Acc. Responses 0 Tx Timeouts
0
Rx Bad Authenticators       0
Rx Unknown Types            0
Rx Packets Dropped          0
State:                      Disabled
Round-Trip Time:            0 ms
```

```

Server #1 (0.0.0.0:1813) RADIUS Accounting Statistics:
Rx Responses                      0 Tx Requests
0
Rx Malformed Responses            0 Tx Retransmissions
0
Rx Bad Authenticators             0 Tx Pending Requests
0
Rx Unknown Types                  0 Tx Timeouts
0
Rx Packets Dropped                0
State:                            Disabled
Round-Trip Time:                  0 ms
Switch(aaa)#

```

2.8 tacacs+

Using this command you can configure the TACACS+ authentication server parameters.

 If no TACACS+ authentication server is configured or reachable, the device will show "Server disconnect".

Syntax:

```

tacacs+ <index> enable [<ip-hostname>] [<0-65535>] [<Line>]
tacacs+ <index> disable

```

Parameter:

<index>

The index of the TACACS+ authentication server.

Possible values: 1 to 5

enable|disable

Enables or disables the TACACS+ authentication for the given index.

<ip-hostname>

The IP address or hostname of the TACACS+ authentication server.

<0-65535>

The UDP port used for connection to TACACS+ authentication server.

Special values: 0 is equivalent to default port (1813).

<LINE>

Shared secret for accessing the external server.

Possible values: maximum 29 characters.

Example:

```

Switch(aaa)# tacacs+ 1 enable 192.168.2.22 0 tacacs
Server disconnect!

```

2.9 *timeout*

Using this command you can configure the server response timeout.

Syntax:

```
timeout <3-3600>
```

Parameter:

<3-3600>

The Timeout is the maximum time to wait for a reply from a server.

Example:

```
Switch(aaa)# timeout 360  
Switch(aaa)#
```

3 Access Commands

This section shows you how to configure access management table of the switch including HTTP/HTTPS, SNMP, and TELNET/SSH. You can manage the switch over an Ethernet LAN, or over the Internet.

Command	Function
add	Add or modify access management entry
clear	Clear access management statistics
delete	Delete access management entry
mode	Configure the access management mode
show	Show access management information

3.1 *add*

Using this command you can add or modify access management parameters. If the host IP address matches the IP range of an entry and the used protocol matches the permitted protocols of this entry the switch will allow management access.

Syntax:

```
add <1-16> ipv4|ipv6 <start-ip> <end-ip> all|snmp|telnet|web
```

Parameter:**<1-16>**

Index of the access management table entry.

ipv4

IPv4 format address.

ipv6

IPv6 format address.

<start-ip>

Indicates the start IP address for the access management entry.

<end-ip>

Indicates the end IP address for the access management entry.

all

Indicates that the host can access the switch from any interface if the host IP address matches the IP address range provided in the entry.

snmp

Indicates that the host can access the switch from SNMP interface if the host IP address matches the IP address range provided in the entry.

telnet

Indicates that the host can access the switch from TELNET/SSH interface if the host IP address matches the IP address range provided in the entry.

web

Indicates that the host can access the switch from HTTP/HTTPS interface if the host IP address matches the IP address range provided in the entry.

Example:

```
Switch(access)# add 1 ipv4 192.168.1.1 192.168.1.241 all
Switch(access)# show config
Access Management Mode : Disabled

W: WEB/HTTPS
S: SNMP
T: TELNET/SSH
Index Start IP Address          End IP Address          W S T
-----
1      192.168.1.1              192.168.1.241          Y Y Y
Switch(access)#
```

3.2 clear

Using this command you can clear access management statistics.

Syntax:

```
clear statistics
```

Parameter:**statistics**

Clear access management statistics

Example:

```
Switch(access)# clear statistics
Switch(access)#
```

3.3 delete

Using this command you can delete an access management entry.

Syntax:

```
delete <1-16>
```

Parameter:**<1-16>**

Index of the entry to be deleted.

Example:

```
Switch(access)# delete 1
Switch(access)# show config
Access Management Mode : Disabled
W: WEB/HTTPS
S: SNMP
T: TELNET/SSH
Index Start IP Address          End IP Address          W S T
-----
Switch(access)#
```

3.4 mode

Using this command you can configure the access management mode.

Syntax:

```
mode disable|enable
```

Parameter:**disable**

Disables access management mode.

enable

Enables access management mode.

Example:

```
Switch(access)# mode enable
Switch(access)#
Switch(access)# show config
Access Management Mode : Enabled
W: WEB/HTTPS
S: SNMP
T: TELNET/SSH
Index Start IP Address          End IP Address          W S T
-----
1      192.168.2.22              192.168.2.250          Y Y Y
Switch(access)#
```

3.5 show

Using this command you can display access configuration and statistics.

Syntax:

```
show config|statistics
```

Parameter:**config**

Shows current access management configuration.

statistics

Shows available access management statistics.

Example:

```
Switch(access)# show config

Access Management Mode : Enabled

W: WEB/HTTPS
S: SNMP
T: TELNET/SSH
Index Start IP Address          End IP Address          W S T
-----
UDP Port

Switch(access)# show statistics
Client  Receive    Allow      Discard
-----
HTTP    0           0           0
HTTPS   0           0           0
SNMP    0           0           0
TELNET  0           0           0
SSH     0           0           0
```

4 Account Commands

In this function, only administrator can create, modify or delete the username and password. Administrator can modify other guest identities' password without confirming the password but it is necessary to modify the administrator-equivalent identity. Guest-equivalent identity can modify his password only. Please note that you must confirm administrator/guest identity in the field of Authorization in advance before configuring the username and password. Only one administrator is allowed to exist and unable to be deleted. In addition, up to 4 guest accounts can be created.

Command	Function
add	Add or modify user account
delete	Delete user account
show	Show user account information

4.1 add

Using this command you can add or modify user accounts.

Syntax:

```
add <1-15> <name> [<password>]
```

Parameter:

<1-15>

User privilege level, see [Privilege level Commands](#) on page 223.

<name>

Up to 32 characters to identify the user name.

If the user name is NOT already existing in the account table, an account will be created with the selected name.

If the user name is already existing in the account table, the selected account will be updated with the given parameters



If you omit the password for an existing account, the password will be removed.

<password>

The password for this user name.

Example:

```
Switch(account)# add 10 david david
Switch(account)# show
User Name                Privilege Level
-----
admin                    15
david                    10
```

4.2 *delete*

Using this command you can delete a user account.

Syntax:

```
delete <name>
```

Parameter:

<name>

Up to 32 characters to identify the user name

Example:

```
Switch(account)# delete guest
Switch(account)# show
User Name                Privilege Level
-----
admin                    15
Switch(account)#
```

4.3 *show*

Using this command you can display current user account information.

Syntax:

```
show
```

Example:

```
Switch(account)# show
User Name                Privilege Level
-----
admin                    15
Switch(account)#
```

5 ACL Commands

The switch access control list (ACL) is probably the most commonly used object in the IOS. It is used for packet filtering but also for selecting types of traffic to be analyzed, forwarded, or influenced in some way. The ACLs are divided into EtherTypes, IPv4, ARP protocol, MAC and VLAN parameters etc. Here we will just go over the standard and extended access lists for TCP/IP. As you create ACEs for ingress classification, you can assign a policy for each port, the policy number is 1-8, however, each policy can be applied to any port. This makes it very easy to determine what type of ACL policy you will be working with.

Command	Function
ace	Add or modify access control entry
action	Configure ACL port default action
Clear	Clear all ACL counters
delete	To delete the ACE (Access Control Entry) configuration on the switch
logging	Configure ACL port default logging operation
move	Move ACE
policy	Configure ACL port policy
rate-limiter	To set ACL rate limit
show	Show ACL information
shutdown	Configure ACL port default shut down operation

5.1 ace

Using this command you can add or modify Access Control Entry.

Syntax:

```
ace <1-256> <0-256> ((port <port-list> policy <0-255> <0x00-0xFF>)|switch)
(any|arp|etype|icmp|ipv4|tcp|udp)
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

1st level

<1-256>

If the ACE ID is specified and an entry with this ACE ID already exists, the ACE will be modified. Otherwise, a new ACE will be added.

<0-256>

If the next ACE ID is non-zero, the ACE will be placed before this ACE in the list. If the next ACE ID is zero, the ACE will be placed last in the list.

port

Port ACE keyword, the rule applies to the specified port only.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

policy

Policy ACE keyword, the rule applies to all ports configured with the specified policy.

<0-255>

Policy number.

<0x00-0xFF>

Policy number bitmask.

switch

Switch ACE keyword, the rule applies to all ports

any

Any frame can match this ACE.

arp

Only ARP frames can match this ACE. Notice the ARP frames won't match the ACE with Ethernet type

etype

Only Ethernet Type frames can match this ACE

icmp

Only ICMP frames can match this ACE. Notice the ICM frames won't match the ACE with Ethernet type

ipv4

Only IPv4 frames can match this ACE. Notice the IPv4 frames won't match the ACE with Ethernet type

tcp

Only TCP frames can match this ACE. Notice the TCP frames won't match the ACE with Ethernet type

udp

Only UDP frames can match this ACE. Notice the UDP frames won't match the ACE with Ethernet type

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# show
ACE ID       : 5                      Rate Limiter : Disabled
Ingress Port : 10                    Port Redirect: Disabled
Policy/Bitmask: 7/0xff               Mirror       : Disabled
Type         : User                  Logging      : Disabled
Frame Type   : Any                  Shutdown     : Disabled
Action       : Permit               Counter      : 0

MAC Parameters                                VLAN Parameters
-----
Switch(acl/ace-specific(any))# end
Success! ACE ID 5 added last
```

5.1.1 *action*

Using this command you can configure the forwarding action of the ACE.

Syntax:

```
action deny [port-redirect (<port-list>|disable)] | permit
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

deny

Deny forwarding.

permit

Permit forwarding.

port-redirect

Enables redirect function.

<port-list>

Port list for redirect of frames.

disable

Disables redirect function.

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# action deny port-redirect disable
Switch(acl/ace-specific(any))# end
```

5.1.2 *arp-opcode*

Using this command you can configure the ARP opcode of the ACE.

Syntax:

```
arp-opcode any|arp|other|rarp
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

any

Don't care

arp

Frame must have ARP/RARP opcode set to ARP

other

Frame has unknown ARP/RARP opcode

rarp

Frame must have ARP/RARP opcode set to RARP

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# arp-opcode arp
Switch(acl/ace-specific(any))# end
```

5.1.3 arp-flags

Using this command you can configure the ARP flags of the ACE..

Syntax:

```
arp-flags (arp-smac|ethernet|ip|length|rarp-dmac|request-reply) (<0-1>|any)
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**arp-smac**

Specify whether frames can hit the action according to their sender hardware address field (SHA) settings

ethernet

Specify whether frames can hit the action according to their ARP/RARP protocol address space (PRO) settings

ip

Specify whether frames can hit the action according to their ARP/RARP hardware address space (HRD) settings

length

Specify whether frames can hit the action according to their ARP/RARP hardware address length (HLN) and protocol address length (PLN) settings

rarp-dmac

Specify whether frames can hit the action according to their target hardware address field (THA) settings

request-reply

Specify the available ARP/RARP opcode (OP) flag for this ACE

<0-1>

[arp-smac] 0: ARP frames where SHA is not equal to the SMAC address

[arp-smac] 1: ARP frames where SHA is equal to the SMAC address

[ethernet] 0: ARP/RARP frames where the PRO is not equal to IP (0x800)

[ethernet] 1: ARP/RARP frames where the PRO is equal to IP(0x800)

[ip] 0: ARP/RARP frames where the HLD is not equal to Ethernet(1)

[ip] 1: ARP/RARP frames where the HLD is equal to Ethernet(1)

[length] 0: ARP/RARP frames where the HLN is not equal to Ethernet(0x06) or the (PLN) is not equal to IPv4(0x04)

[length] 1: ARP/RARP frames where the HLN is equal to Ethernet (0x06) and the (PLN) is equal to IPv4(0x04)

[rarp-dmac] 0: RARP frames where THA is not equal to the DMAC address

[rarp-dmac] 1: RARP frames where THA is equal to the DMAC address

[request-reply] 0: Frame must have ARP Reply or RARP Reply OP flag

[request-reply] 1: Frame must have ARP Request or RARP Request OP flag set

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# arp-flags ethernet 1
Switch(acl/ace-specific(any))# end
```

5.1.4 *dip*

Using this command you can configure the destination IP address of the ACE.

Syntax:

```
dip any | (<ip-address> [<1-32>])
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<ip-address>**

A frame that hits this ACE matches this destination IP address value

<1-32>

Network prefix

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# dip any
Switch(acl/ace-specific(any))# end
```

5.1.5 *dmac*

Using this command you can configure the destination MAC address of the ACE.

Syntax:

```
dmac any | broadcast | multicast | unicast
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**any**

Don't care

broadcast

Frame must be broadcast

multicast

Frame must be multicast

unicast

Frame must be unicast

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# dmac broadcast
Switch(acl/ace-specific(any))# end
```

5.1.6 *dport*

Using this command you can configure the destination UDP/TCP port range of the ACE.

Syntax:

```
dport <dport-range>|any
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<dport-range>**

Destination UDP/TCP port range, format: 1,3-5

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# dport 1-5
Switch(acl/ace-specific(any))# end
```

5.1.7 *end*

Using this command you can complete the current command sequence.

Syntax:

```
end <cr>
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<cr>**

No additional parameters required.

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# end
```

5.1.8 *etype*

Using this command you can configure the ethernet type of the ACE.

Syntax:

```
etype <0x0600-0xffff>|any
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

<0x0600-0xffff>

A frame that hits this ACE matches this EtherType value

any

Any but excluding 0x0800(IPv4) 0x0806(ARP) and 0x86DD(IPv6)

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# etype any
Switch(acl/ace-specific(any))# end
```

5.1.9 *icmp-code*

Using this command you can configure the ICMP code of the ACE.

Syntax:

```
icmp-code <0-255>|any
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

<0-255>

A frame that hits this ACE matches this ICMP code value

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# icmp-code 55
Switch(acl/ace-specific(any))# end
```

5.1.10 *icmp-type*

Using this command you can configure the ICMP type of the ACE.

Syntax:

```
icmp-type <0-255> | any
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<0-255>**

A frame that hits this ACE matches this ICMP type value

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# icmp-type 5
Switch(acl/ace-specific(any))# end
```

5.1.11 *ip-flags*

Using this command you can configure the IP flags of the ACE.

Syntax:

```
ip-flags (fragment|options|ttl) (<0-1> | any)
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**fragment**

Specify the fragment offset settings for this ACE

<0-1>

[fragment] 0: IPv4 frames where the MF bit is unset and the FRAG OFFSET field is equal zero

[fragment] 1: IPv4 frames where the MF bit is set or the FRAG OFFSET field is greater than zero

[options] 0: IPv4 frames where the options flag is unset

[options] 1: IPv4 frames where the options flag is set

[ttl] 0: IPv4 frames with a Time-to-Live field is equal to zero

[ttl] 1: IPv4 frames with a Time-to-Live field greater than zero

any

Don't care

options

Specify the IP options flag setting for this ACE

ttl

Specify the Time-to-Live settings for this ACE

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# ip-flags ttl 0
Switch(acl/ace-specific(any))# end
```

5.1.12 *ip-protocol*

Using this command you can configure the IP protocol of the ACE.

Syntax:

```
ip-protocol <0-255>|any
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<0-255>**

A frame that hits this ACE matches this IP protocol value

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# ip-protocol 77
Switch(acl/ace-specific(any))# end
```

5.1.13 *logging*

Using this command you can configure the logging operation of the ACE.

Syntax:

```
logging <port-list> enable|disable
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**disable**

Frames matching the ACE are not logged

enable

Frames matching the ACE are stored in the system log

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# logging enable
Switch(acl/ace-specific(any))# end
```

5.1.14 *mirror*

Using this command you can configure the mirror operation of the ACE.

Syntax:

```
mirror <port-list> disable|enable
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**disable**

Frames matching the ACE are not mirrored

enable

Frames matching the ACE are mirrored

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# mirror disable
Switch(acl/ace-specific(any))# end
```

5.1.15 *rate*

Using this command you can configure the rate limit of the ACE.

Syntax:

```
rate <1-16>|disable
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<1-16>**

Rate limiter ID

disable

Rate limit is disabled for the ACE

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# rate disable
Switch(acl/ace-specific(any))# end
```

5.1.16 *show*

Using this command you can show the current ACE settings.

Syntax:

```
show <cr>
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

<cr>

No parameters required.

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# show
Switch(acl/ace-specific(any))# end
```

5.1.17 *shutdown*

Using this command you can configure the port shut down operation for the ACE.

Syntax:

```
shutdown disable|enable
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

disable

Port shut down is disabled for the ACE

enable

If a frame matches the ACE, the ingress port will be disabled

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# shutdown enable
Switch(acl/ace-specific(any))# end
```

5.1.18 *sip*

Using this command you can configure the source IP address of the ACE.

Syntax:

```
sip any|(<ip-address> [<1-32>])
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<ip-address>**

A frame that hits this ACE matches this destination IP address value

<1-32>

Network prefix

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# sip any
Switch(acl/ace-specific(any))# end
```

5.1.19 *smac*

Using this command you can configure the source MAC address for the ACE.

Syntax:

```
smac <mac-address>|any
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<mac-address>**

A frame that hits this ACE matches this source MAC address value

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# mac-address any
Switch(acl/ace-specific(any))# end
```

5.1.20 *sport*

Using this command you can configure the source UDP/TCP port for the ACE.

Syntax:

```
sport <sport-range>|any
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<sport-range>**

Source UDP/TCP port range

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# sport any
Switch(acl/ace-specific(any))# end
```

5.1.21 *tcp-flags*

Using this command you can configure the TCP flags for the ACE.

Syntax:

```
tcp-flags      ack|fin|psh|rst|syn|urg (<0-1>|any)
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**ack**

Specify the TCP "Acknowledgment field significant" (ACK) value for this ACE

fin

Specify the TCP "No more data from sender" (FIN) value for this ACE

psh

Specify the TCP "Push Function" (PSH) value for this ACE

rst

Specify the TCP "Reset the connection" (RST) value for this ACE

syn

Specify the TCP "Synchronize sequence numbers" (SYN) value for this ACE

urg

Specify the TCP "Urgent Pointer field significant" (URG) value for this ACE

<0-1>

0: TCP frames where the ack|fin|psh|rst|syn|urg field is unset

1: TCP frames where the ack|fin|psh|rst|syn|urg field is set

any

Don't care

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# tcp-flags ack 1
Switch(acl/ace-specific(any))# end
```

5.1.22 *vlan*

Using this command you can configure VLAN parameters of the ACE.

Syntax:

```
vlan (priority (any|<0-7>)) | (tagged any|disable|enable) | (vid any|<1-4094>)
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**priority****any**

Don't care

<0-7>

A frame that hits this ACE matches this tag priority

Specify the tag priority for this ACE

tagged

Specify whether frames can hit the action according to the 802.1Q tagged

any

Don't care

disable

Untagged frame only

enable

Tagged frame only

vid

Specify the VLAN ID filter for this ACE

any

No VLAN ID filter is specified. (VLAN ID filter status is don't-care.)

<1-4094>

A frame that hits this ACE matches this VLAN ID value

Example:

```
Switch(acl)# ace 5 0 port 10 policy 7 0xFF any
Switch(acl/ace-specific(any))# vlan priority 5
Switch(acl/ace-specific(any))# end
```

5.2 *action*

Using this command you can configure ACL port default action.

Syntax:

```
action <port-list> deny [port-redirect (<port-list>|disable)]
action <port-list> permit
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

deny

Deny forwarding

permit

Permit forwarding

port-redirect

port redirect keyword

<port-list>

Port list for copy of frames

disable

Disables port copy

Example:

```
Switch(acl)# action 1 permit
Switch(acl)#
Switch(acl)# show port
                        Rate
Port Policy Action Limiter Port Copy   Mirror   Logging Shutdown Counter
-----
1      1      Deny   Disabled Disabled      Disabled Disabled Disabled 0
2      1      Permit Disabled Disabled      Disabled Disabled Disabled 0
3      1      Permit Disabled Disabled      Disabled Disabled Disabled 0
4      1      Permit Disabled Disabled      Disabled Disabled Disabled 0
5      1      Permit Disabled Disabled      Disabled Disabled Disabled 0
.....
Rate Limiter   Rate
-----
1              1 PPS
2              1 PPS
3              1 PPS
4              1 PPS
```

5.3 clear

Clear all ACL counters.

Syntax:

```
clear
```

Example:

```
Switch(acl)# clear
Switch(acl)#
```

5.4 *delete*

Using this command you can delete the ACE (Access Control Entry) configuration on the switch.

Syntax:

```
delete <1-256>
```

Parameter:

<1-256>

ACE ID must be exist

Example:

```
Switch(acl)# delete 1
Switch(acl)#
Switch(acl)# show acl-config
Number of ACEs: 0
```

5.5 *logging*

Using this command you can configure ACL port default logging operation.

Syntax:

```
logging <port-list> enable|disable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Frames received on the port are not logged

enable

Frames received on the port are stored in the system log

Example:

```
Switch(acl)# logging 1 disable
Switch(acl)#
```

5.6 *mirror*

Configure ACL port default mirror operation.

Syntax:

```
mirror <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Frames received on the port are not mirrored

enable

Frames received on the port are mirrored

Example:

```
Switch(acl)# mirror 5 enable
Switch(acl)#
```

5.7 *move*

Using this command you can move ACE configuration between two indexes.

Syntax:

```
move <1-256> <0-256>
```

Parameter:

<1-256>

ACE ID must be exist

<0-256>

If the next ACE ID is non-zero, the ACE will be Placed before this ACE in the list. If the next ACE ID is zero, the ACE will be placed last in the list.

Example:

```
Switch(acl)# move 1 0
Switch(acl)#
```

5.8 *policy*

Using this command you can set acl port policy on switch.

Syntax:

```
policy <port-list> <0-255>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-255>

Policy number

Example:

```
Switch(acl)# policy 1 1
Switch(acl)#
```

5.9 *port-rate*

Using this command you can set acl port-rate on switch.

Syntax:

```
port-rate <port-list> (<1-16>|disable)
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<1-16>

Rate limiter ID

disable

Disables rate limit

Example:

```
Switch(acl)# port-rate 1 1
Switch(acl)#
```

5.10 *rate-limiter*

Using this command you can set the access control rule with rate limiter on switch.

Syntax:

```
rate-limiter <1-16> kbps <0-10000>  
rate-limiter <1-16> pps <0-3276700>
```

Parameter:

<1-16>

Rate limiter ID

kbps

Kbits per second

<0-10000>

Rate in 100Kbps (the value 1 correlates to 100 kbps).

pps

Packets per second

<0-3276700>

Rate in pps (the value 1 correlates to 1 pps).

Example:

```
Switch(acl)# rate-limiter 1 kbps 100  
Switch(acl)#
```

5.11 *show*

Using this command you can show all access control entry setting or information of the switch.

Syntax:

```
show acl-config [<1-256>]  
show acl-status|port|rate-limiter
```

Parameter:

acl-config

Shows ACL configuration

<1-256>

Shows detail ACE configuration by ACE ID

acl-status

Shows ACL status

port

Shows ACL port configuration

rate-limiter

Shows ACL rate limiter

Example:

```
Switch(acl)# show acl-config
Number of ACEs: 0

Switch(acl)# show port
Rate
Port Policy Action Limiter Port Copy Mirror Logging Shutdown Counter
-----
1 1 Permit 1 Disabled Disabled Disabled Disabled 0
2 1 Permit Disabled Disabled Disabled Disabled Disabled 0
3 1 Permit Disabled Disabled Disabled Disabled Disabled 0
4 1 Permit Disabled Disabled Disabled Disabled Disabled 0
5 1 Permit Disabled Disabled Disabled Disabled Disabled 0
6 1 Permit Disabled Disabled Disabled Disabled Disabled 0
7 1 Permit Disabled Disabled Disabled Disabled Disabled 0
8 1 Permit Disabled Disabled Disabled Disabled Disabled 0
9A 1 Permit Disabled Disabled Disabled Disabled Disabled 0
10A 1 Permit Disabled Disabled Disabled Disabled Disabled 0
9B 1 Permit Disabled Disabled Disabled Disabled Disabled 0
10B 1 Permit Disabled Disabled Disabled Disabled Disabled 0

Rate Limiter Rate
-----
1 1 PPS
2 1 PPS
3 1 PPS
4 1 PPS
5 1 PPS
--More--, q to quit
```

5.12 shutdown

Configure ACL port default shut down operation.

Syntax:

```
shutdown <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Port shut down is disabled.

enable

If a frame is received on the port, the port will be disabled.

Example:

```
Switch(acl)# shutdown 5 disable
Switch(acl)#
```

5.13 *state*

Configure ACL port state.

Syntax:

```
state <port-list>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables ACL port state

enable

Enables ACL port state

Example:

```
Switch(acl)# state 5 enable
Switch(acl)#
```

6 Aggregation Commands

The Aggregation is used to configure the settings of Link Aggregation. You can bundle more than one port with the same speed, full duplex and the same MAC to be a single logical port, thus the logical port aggregates the bandwidth of these ports. This means you can apply your current Ethernet equipment's to build the bandwidth aggregation. For example, if there are three Fast Ethernet ports aggregated in a logical port, then this logical port has bandwidth three times as high as a single Fast Ethernet port has.

Command	Function
delete	Delete command
group	Configure the link aggregation group
mode	Configure the link aggregation traffic distribution mode
Show	Show aggregation group information

6.1 *delete*

Using this command you can delete a selected link aggregation group.

Syntax:

```
delete group <group-id>
```

Parameter:

group

Delete instruction for groups.

group-id

Group ID you want to delete.

Example:

```
Switch(aggregation)# delete group 2
Switch(aggregation)# show
Aggregation Mode
-----
Source MAC      : Disabled
Destination MAC : Disabled
IP Address      : Disabled
TCP/UDP Port    : Disabled
```

6.2 *group*

Using this command you can define a selected link aggregation group.

Syntax:

```
group <1-13> <port-list>
```

Parameter:**<1-13>**

The Aggregation group id.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(aggregation)# group 2 5-7
Switch(aggregation)#
```

6.3 mode

Using this command you can enable or disable the link aggregation for different criteria.

enable or disable

Syntax:

```
mode (dmac|ip|port|smac) (disable|enable)
```

Parameter:**dmac**

Destination MAC address.

ip

Source and destination IP address.

port

Source and destination UDP/TCP port

smac

Source MAC address

disable

Disables field in traffic distribution

enable

Enables field in traffic distribution

Example:

```
Switch(aggregation)# mode ip disable
Switch(aggregation)#
Switch(aggregation)# show
Aggregation Mode
-----
```

```
Source MAC      : Disabled
Destination MAC : Disabled
IP Address      : Disabled
TCP/UDP Port    : Disabled

Group ID  Name    Type    Configured Ports  Aggregated Ports
-----  -
2         LLAG2   Static  5-7               None
Switch(aggregation)#
```

6.4 *show*

Using this command you can show the current link aggregation configuration.

Syntax:

show

Example:

```
Switch(aggregation)# show
Aggregation Mode
-----
Source MAC      : Enabled
Destination MAC : Disabled
IP Address      : Disabled
TCP/UDP Port    : Enabled

Group ID  Name    Type    Configured Ports  Aggregated Ports
-----  -
2         LLAG2   Static  5-7               None
Switch(aggregation)#
```

7 Arp-inspection Commands

The section describes how to configure the ARP inspection parameters of the switch. You could use the ARP inspection configuration to manage the ARP table.

Command	Function
add	Add ARP inspection static entry
delete	Delete ARP inspection static entry
mode	Configure ARP inspection mode
port-mode	Configure ARP inspection port mode
show	Show ARP inspection information

7.1 add

Using this command you can add a static entry in ARP inspection table.

Syntax:

```
add <port-list> <1-4094> <ip-address> <mac-address>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<1-4094>

VLAN ID, possible values from 1 to 4094.

<ip-address>

IP address allowed for doing ARP request

<mac-address>

MAC address, format 0a-1b-2c-3d-4e-5f

Example:

```
Switch(arp-inspection)# add 1 5 192.168.1.2 0a-1b-2c-3d-4e-5f
Switch(arp-inspection)#
```

7.2 delete

Using this command you can delete a static entry in ARP inspection table.

Syntax:

```
delete <port-list> <1-4094> <ip-address> <mac-address>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<1-4094>

VLAN ID, possible values from 1 to 4094.

<ip-address>

IP address allowed for doing ARP request

<mac-address>

MAC address, format 0a-1b-2c-3d-4e-5f

Example:

```
Switch(arp-inspection)# delete 1 5 192.168.1.2 0a-1b-2c-3d-4e-5f
Switch(arp-inspection)#
```

7.3 mode

Using this command you can configure the ARP inspection mode.

Syntax:

```
mode disable|enable
```

Parameter:**disable**

Globally disable ARP inspection mode.

enable

Globally enable ARP inspection mode.

Example:

```
Switch(arp-inspection)# mode disable
Switch(arp-inspection)#
```

7.4 port-mode

Using this command you can configure the ARP inspection port mode.

Syntax:

```
port-mode <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables ARP inspection port mode

enable

Enables ARP inspection port mode

Example:

```
Switch(arp-inspection)# port-mode 1 disable
Switch(arp-inspection)#
```

7.5 show

Using this command you can show the current ARP inspection configuration.

Syntax:

```
show config|status
```

Parameter:**config**

Shows ARP inspection configuration

status

Shows ARP inspection static and dynamic entry

Example:

```
Switch(arp-inspection)# show config

ARP Inspection Mode : Disabled

Port  Port Mode
----  -
1     Disabled
2     Disabled
3     Disabled
4     Disabled
5     Disabled
6     Disabled
7     Disabled
8     Disabled
9     Disabled
10    Disabled
```

```
11 Disabled
12 Disabled
13 Disabled
14 Disabled
15 Disabled
16 Disabled
17 Disabled
18 Disabled
19 Disabled
20 Disabled
21 Disabled
22 Disabled
23 Disabled
24 Disabled
25 Enabled
26 Disabled
27 Disabled
28 Disabled
Switch(arp-inspection)#
```

7.6 *translate*

Using this command you can translate dynamic ARP inspection entries into static entries.

Syntax:

```
translate
```

Example:

```
Switch(arp-inspection)# translate
ARP Inspection:
    Translate 0 dynamic entries into static entries.
Switch(arp-inspection)#
```

8 Auth Method Commands

The section describes how to configure the local authentication method.

Command	Function
fallback	Configure local authentication fallback
method	Configure authentication method
show	Show Authentication configuration

8.1 *fallback*

Using this command you can configure the local authentication fallback function.

Syntax:

```
fallback (console|ssh|telnet|web) (disable|enable)
```

Parameter:

console

Settings the authenticate method fallback via console

ssh

Settings the authenticate method fallback via ssh

telnet

Settings the authenticate method fallback via telnet

web

Settings the authenticate method fallback via web

disable

Disables local authentication if remote authentication fails

enable

Enables local authentication if remote authentication fails

Example:

```
Switch(auth)# fallback ssh disable
Switch(auth)#
```

8.2 *method*

Using this command you can configure the local authentication method.

Syntax:

```
method (console|ssh|telnet|web) (local|none|radius|tacacs+)
```

Parameter:**console**

Settings the authenticate method via console

ssh

Settings the authenticate method via ssh

telnet

Settings the authenticate method via telnet

web

Settings the authenticate method via web

local

Use local authentication

none

Authentication disabled

telnet

Use remote RADIUS authentication

tacacs+

Use remote TACACS+ authentication

Example:

```
Switch(auth)# method ssh local
Switch(auth)#
```

8.3 show

Using this command you can show the current authentication method configuration.

Syntax:

```
show
```

Example:

```
Switch(auth)# show
Client      Authentication Method  Local Authentication Fallback
-----
console     local                  Disabled
telnet      local                  Disabled
ssh         local                  Disabled
web         local                  Disabled10B   Disabled
```

9 Config-file Commands

This section describes how to export and import the Switch configuration. Any current configuration files will be exported as XML format.

Command	Function
export	Export configuration file to TFTP server
import	Import configuration file from TFTP server

9.1 *export*

Using this command you can export the current device configuration to a TFTP server.

Syntax:

```
export <ip-address> [<filename>]
```

Parameter:

<ip-address>

The TFTP server ip address

<filename>

Configuration file name.

Example:

```
Switch(config-file)# export 192.168.1.100 testfile
Switch(config-file)#
```

9.2 *import*

Using this command you can import a device configuration from a TFTP server.

Syntax:

```
import <ip-address> <filename> [check]
```

Parameter:

<ip-address>

The TFTP server ip address

<filename>

Configuration file name.

check

Check configuration file only, do not start import.

Example:

```
Switch(config-file)# import 192.168.1.100 testfile
Switch(config-file)#
```

10 DHCP Relay Commands

The section describes how to forward DHCP requests to another specific DHCP server via DHCP relay. The DHCP servers may be on another network.

Command	Function
clear	Clear DHCP relay statistics
mode	Configure DHCP relay mode
relay-option	Configure DHCP relay agent information option
server	Configure DHCP relay server
show	Show DHCP relay information

10.1 *clear*

Using this command you can clear the DHCP relay statistics.

Syntax:

```
clear statistics
```

Parameter:

statistics

The parameter let you to clear DHCP relay statistics.

Example:

```
Switch(dhcp-relay)# clear statistics
Switch(dhcp-relay)#
```

10.2 *mode*

Using this command you can configure the DHCP relay mode.



When enable DHCP relay mode operation, the agent forward and to transfer DHCP messages between the clients and the server when they are not on the same subnet domain. And the DHCP broadcast message won't flood for security considered.

Syntax:

```
mode disable|enable
```

Parameter:**disable**

Disables DHCP relay mode.

enable

Enables DHCP snooping mode.

Example:

```
Switch(dhcp-relay)# mode disable
Switch(dhcp-relay)#
```

10.3 *relay-option*

Using this command you can configure the DHCP relay agent option.

 The agent insert specific information (option 82) into a DHCP message when forwarding to DHCP server and remove it from a DHCP message when transferring to DHCP client. If agent receive a DHCP message that already contains relay agent information. It will enforce the policy.

Syntax:

```
relay-option disable
relay-option enable [drop|keep|replace]
```

Parameter:**disable**

The parameter means you to disable DHCP relay agent information option mode.

enable

The parameter means you to enable DHCP relay agent information option mode.

drop

Drop the packet when receive a DHCP message that already contains relay information.

keep

Keep the original relay information when receive a DHCP message that already contains it.

replace

Replace the original relay information when receive a DHCP message that already contains it.

Example:

```
Switch(dhcp-relay)# relay-option disable
Switch(dhcp-relay)#
```

10.4 *server*

Using this command you can configure the DHCP relay server.

Syntax:

```
server <ip-address>
```

Parameter:**<ip-address>**

The parameter let you type in the DHCP server IP address.

Example:

```
Switch(dhcp-relay)# server 192.168.1.100
Switch(dhcp-relay)# show config
DHCP Relay Mode           : Disabled
DHCP Relay Server         : 192.168.1.100
DHCP Relay Information Mode : Disabled
DHCP Relay Information Policy : Replace
Switch(dhcp-relay)#
```

10.5 show

Using this command you can show the current DHCP relay configuration.

Syntax:

```
show config|statistics
```

Parameter:**config**

The parameter lets you to set for show DHCP relay configuration

statistics

The parameter lets you to set for show DHCP relay statistics

Example:

```
Switch(dhcp-relay)# show config
DHCP Relay Mode           : Disabled
DHCP Relay Server         : 192.168.1.100
DHCP Relay Information Mode : Disabled
DHCP Relay Information Policy : Replace
Switch(dhcp-relay)# show statistics

Server Statistics:
-----
Transmit to Server      :      0 Transmit Error          :
0
Receive from Server     :      0 Receive Missing Agent Option :
0
Receive Missing Circuit ID :      0 Receive Missing Remote ID   :
0
Receive Bad Circuit ID   :      0 Receive Bad Remote ID     :
0
```

```
Client Statistics:
-----
Transmit to Client   :      0   Transmit Error       :      0
Receive from Client  :      0   Receive Agent Option  :      0
Replace Agent Option :      0   Keep Agent Option   :      0
Drop Agent Option    :      0
Switch(dhcp-relay)#
```

11 DHCP Snooping Commands

The section describes to configure the DHCP Snooping parameters of the switch. The DHCP Snooping can prevent attackers from adding their own DHCP servers to the network.

Command	Function
clear	Clear DHCP snooping statistics
mode	Configure DHCP snooping mode
Port-mode	Configure DHCP snooping port mode
show	Show DHCP snooping information

11.1 *clear*

Using this command you can clear the DHCP snooping statistics.

Syntax:

```
clear statistics <port-list>
```

Parameter:

statistics

The parameter let you to clear DHCP snooping statistics.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(dhcp-snooping)# clear statistics 1
Switch(dhcp-snooping)#
```

11.2 *mode*

Using this command you can configure the DHCP snooping mode.



When enable DHCP snooping mode operation, the request DHCP messages will be forwarded to trusted ports and only allowed reply packets from trusted ports.

Syntax:

```
mode disable|enable
```

Parameter:**disable**

Disables DHCP snooping mode.

enable

Enables DHCP snooping mode.

Example:

```
Switch(dhcp-snooping)# mode disable
Switch(dhcp-snooping)#
```

11.3 *port-mode*

Using this command you can configure the DHCP relay port mode.

Syntax:

```
port-mode <port-list> trusted|untrusted
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

trusted

Configures the port as trusted source of the DHCP message

untrusted

Configures the port as untrusted source of the DHCP message

Example:

```
Switch(dhcp-snooping)# port-mode 1 trusted
Switch(dhcp-snooping)#
Switch(dhcp-snooping)# show config

DHCP Snooping Mode : Disabled
Port  Port Mode
----  -
1      trusted
2      untrusted
3      untrusted
4      untrusted
5      untrusted
6      untrusted
7      untrusted
8      untrusted
9      untrusted
10     untrusted
11     untrusted
12     untrusted
13     untrusted
14     untrusted
```

```
15    untrusted
16    untrusted
17    untrusted
18    untrusted
--More--, q to quit
Switch(dhcp-snooping)#
```

11.4 *show*

Using this command you can show the current DHCP snooping configuration.

Syntax:

```
show config
show statistics <port-id>
```

Parameter:

config

Shows DHCP snooping configuration.

statistics

Shows DHCP snooping statistics.

<port-id>

ID of the port the statistics shall be displayed for.

Example:

```
Switch(dhcp-snooping)# port-mode 1 trusted
Switch(dhcp-snooping)#
Switch(dhcp-snooping)# show config

DHCP Snooping Mode : Disabled
Port  Port Mode
----  -
1     trusted
2     untrusted
3     untrusted
4     untrusted
5     untrusted
6     untrusted
7     untrusted
8     untrusted
9     untrusted
10    untrusted
11    untrusted
12    untrusted
13    untrusted
14    untrusted
15    untrusted
16    untrusted
17    untrusted
18    untrusted
```

```
--More--, q to quit
Switch(dhcp-snooping)#
Switch(dhcp-snooping)# show statistics 1
Port 1 Statistics:          Receive Packets          Transmit
Packets
-----
Rx Discover                  0 Tx Discover
  0
Rx Offer                     0 Tx Offer
  0
Rx Request                   0 Tx Request
  0
Rx Decline                   0 Tx Decline
  0
Rx ACK                       0 Tx ACK
  0
Rx NAK                       0 Tx NAK
  0
Rx Release                   0 Tx Release
  0
Rx Inform                    0 Tx Inform
  0
Rx Lease Query               0 Tx Lease Query
  0
Rx Lease Unassigned          0 Tx Lease Unassigned
  0
Rx Lease Unknown             0 Tx Lease Unknown
  0
Rx Lease Active              0 Tx Lease Active
  0
Switch(dhcp-snooping)#
```

12 Diagnostic Commands

This section provides a set of basic system diagnosis. It let users know that whether the system is health or needs to be fixed. The basic system check includes ICMP Ping, ICMPv6, and VeriPHY Cable Diagnostics.

Command	Function
ping	Uses the ICMP protocol's mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway.
ping6	Uses the ICMP protocol's mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway.
veriphy	Run cable diagnostics.

12.1 *ping*

Using this command you can the ICMP protocol's ECHO_REQUEST datagram to request an ICMP ECHO_RESPONSE from a host or gateway.

Syntax:

```
ping <ip-hostname> [<2-1452>] [<1-60>] [<0-30>]
```

Parameter:**<ip-hostname>**

IPv4 Hostname or IP address.

<2-1452>

Size of ICMP echo packet.

<1-60>

Count of ICMP echo packet.

<0-30>

Interval of ICMP echo packet.

Example:

```
Switch(diagnostic)# ping 192.168.6.200 80
PING server 192.168.6.200, 80 bytes of data.
88 bytes from 192.168.6.200: icmp_seq=0, time=0ms
88 bytes from 192.168.6.200: icmp_seq=1, time=0ms
88 bytes from 192.168.6.200: icmp_seq=2, time=0ms
88 bytes from 192.168.6.200: icmp_seq=3, time=0ms
88 bytes from 192.168.6.200: icmp_seq=4, time=0ms
Sent 5 packets, received 5 OK, 0 bad
Switch(diagnostic)#
```

12.2 *ping6*

Using this command you can the ICMP protocol's ECHO_REQUEST datagram to request an ICMP ECHO_RESPONSE from a host or gateway.

Syntax:

```
ping6 <ipv6-address> [<2-1452>] [<1-60>] [<0-30>]
```

Parameter:**<ipv6-address>**

IPv6 Hostname or IP address.

<2-1452>

Size of ICMP echo packet.

<1-60>

Count of ICMP echo packet.

<0-30>

Interval of ICMP echo packet.

Example:

```
Switch(diagnostic)# ping6 ff06:0:0:0:0:0:c3 80
PING6 server ff06::c3, 80 bytes of data.
88 bytes from 192.168.6.200: icmp_seq=0, time=0ms
88 bytes from 192.168.6.200: icmp_seq=1, time=0ms
88 bytes from 192.168.6.200: icmp_seq=2, time=0ms
88 bytes from 192.168.6.200: icmp_seq=3, time=0ms
88 bytes from 192.168.6.200: icmp_seq=4, time=0ms
Sent 5 packets, received 5 OK, 0 bad
Switch(diagnostic)#
```

12.3 *veriphy*

Using this command you can verify the physical cable connection for each port.

Syntax:

```
veriphy <port-list>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(diagnostic)# veriphy 1
Starting VeriPHY, please wait
Port    Pair A    Length    Pair B    Length    Pair C    Length    Pair D    Length
-----
1        OK        255       OK        255       OK        255       OK        255
Switch(diagnostic)#
```

13 Easyport Commands

Easy Port provides a convenient way to save and share common configurations. You can use it to enable features and settings based on the location of a switch in the network and for mass configuration deployments across the network. You could easy to implement included Voice IP phone, Wireless Access Point and IP Camera ... etc. Others you can leverage configuration to run a converged voice, video, and data network considering quality of service (QoS), bandwidth, latency, and high performance.

Command	Function
ip-cam	To set the IP-CAM Configuration on the switch
ip-phone	To set the IP-Phone Configuration on the switch
wifi-ap	To set the WIFI-AP Configuration on the switch.

13.1 *ip-cam*

Using this command you can configure IP-Cam parameters.



The command configuration has level rule, you need to set the port-list what you want to assign setting profile first, and then enter to 2nd level to set every parameters.

Syntax:

```
ip-cam <port-list>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# vlan-mode trunk
Switch(easyport/ip-cam)# access-vlan 8
Switch(easyport/ip-cam)# traffic-class 7
Switch(easyport/ip-cam)# psec-action both
Switch(easyport/ip-cam)# psec-limit 40
Switch(easyport/ip-cam)# psec-mode enable
Switch(easyport/ip-cam)# admin-edge enable
Switch(easyport/ip-cam)# bpdu-guard enable
Switch(easyport/ip-cam)# show
Role                               : IP-CAM
Access VLAN                       : 8
VLAN Mode                         : Trunk
Traffic Class                     : 7
Port Security Mode                : Enabled
Port Security Action              : Trap & Shutdown
Port Security Limit               : 40
STP Admin Edge                   : Enabled
```

```
STP BPDU Guard          : Enabled
Switch(easyport/ip-cam)#
```

13.1.1 *access-vlan*

Using this command you can configure access VLAN for IP Camera.

Syntax:

```
access-vlan          <1-4094>
```

Parameter:

<1-4094>

Access VLAN ID, possible values from 1 to 4094..

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# access-vlan 8
Switch(easyport/ip-cam)#
```

13.1.2 *admin-edge*

Using this command you can configure spanning tree admin-edge for IP Camera.

Syntax:

```
admin-edge          disable|enable
```

Parameter:

disable

Disables spanning tree admin edge.

enable

Enables spanning tree admin edge.

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# admin-edge enable
Switch(easyport/ip-cam)#
```

13.1.3 *bpdu-guard*

Using this command you can configure spanning tree BPDU guard for IP Camera..

Syntax:

```
bpdu-guard          disable|enable
```

Parameter:**disable**

Disables spanning tree BPDU guard.

enable

Enables spanning tree BPDU guard.

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# bpdu-guard enable

Switch(easyport/ip-cam)#
```

13.1.4 *end*

Using this command you can finish IP camera settings and return.

Syntax:

```
end
```

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# end

Switch(easyport/ip-cam)#
```

13.1.5 *psec-action*

Using this command you can configure port security action for IP Camera.

Syntax:

```
psec-action      both|none|shutdown|trap
```

Parameter:**both**

Send a SNMP trap and shutdown the port.

none

Do nothing.

shutdown

Shutdown the port.

trap

Send a SNMP trap.

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# psec-action both
```

```
Switch(easyport/ip-cam)#
```

13.1.6 *psec-limit*

Using this command you can configure port security maximum for IP Camera.

Syntax:

```
psec-limit <1-50>
```

Parameter:

<1-50>

Max. number of MAC addresses.

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# psec-limit 40
Switch(easyport/ip-cam)#
```

13.1.7 *psec-mode*

Using this command you can configure port security mode for IP Camera.

Syntax:

```
psec-mode <1-50>
```

Parameter:

disable

Disables port security.

enable

Enables port security.

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# psec-mode enable
Switch(easyport/ip-cam)#
```

13.1.8 *show*

Using this command you can display current IP camera settings.

Syntax:

```
show
```

Example:

```
Switch(easyport/ip-cam)# show
Role                : IP-CAM
Access VLAN         : 8
VLAN Mode           : Trunk
Traffic Class       : 7
Port Security Mode   : Enabled
Port Security Action : Trap & Shutdown
Port Security Limit  : 40
STP Admin Edge      : Enabled
STP BPDU Guard      : Enabled
```

13.1.9 *traffic-class*

Using this command you can configure traffic class for IP Camera.

Syntax:

```
traffic-class      <0-7>
```

Parameter:

<0-7>

0:Low, 7:High.

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# traffic-class 7

Switch(easyport/ip-cam)#
```

13.1.10 *vlan-mode*

Using this command you can configure VLAN mode for IP Camera.

Syntax:

```
vlan-mode          access|hybrid|trunk
```

Parameter:**access**

Untag all frames.

hybrid

Tag all frames except VLAN ID same as PVID.

trunk

Tag all frames.

Example:

```
Switch(easyport)# ip-cam 22
Switch(easyport/ip-cam)# vlan-mode trunk
```

```
Switch(easyport/ip-cam)#
```

13.2 *ip-phone*

Using this command you can configure IP-Phone parameters.

Syntax:

```
ip-phone <port-list>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
witch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# access-vlan 20
Switch(easyport/ip-phone)# voice-vlan 20
Switch(easyport/ip-phone)# psec-mode enable
Switch(easyport/ip-phone)# psec-limit 30
Switch(easyport/ip-phone)# traffic-class 7
Switch(easyport/ip-phone)# vlan-mode access
Switch(easyport/ip-phone)# psec-action both
Switch(easyport/ip-phone)# save start
Switch(easyport/ip-phone)# show
Role                : IP-Phone
Access VLAN         : 20
VLAN Mode            : Access
Voice VLAN           : 20
Traffic Class        : 7
Port Security Mode   : Enabled
Port Security Action : Trap & Shutdown
Port Security Limit  : 30
STP Admin Edge       : Enabled
STP BPDU Guard       : Enabled

Switch(easyport/ip-phone)#
```

13.2.1 *access-vlan*

Using this command you can configure access VLAN for IP Phone.

Syntax:

```
access-vlan <1-4094>
```

Parameter:

<1-4094>

Access VLAN ID, possible values from 1 to 4094..

Example:

```
Switch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# access-vlan 8

Switch(easyport/ip-phone)#
```

13.2.2 *admin-edge*

Using this command you can configure spanning tree admin-edge for IP Phone.

Syntax:

```
admin-edge          disable|enable
```

Parameter:**disable**

Disables spanning tree admin edge.

enable

Enables spanning tree admin edge.

Example:

```
Switch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# admin-edge enable

Switch(easyport/ip-phone)#
```

13.2.3 *bpdu-guard*

Using this command you can configure spanning tree BPDU guard for IP Phone..

Syntax:

```
bpdu-guard          disable|enable
```

Parameter:**disable**

Disables spanning tree BPDU guard.

enable

Enables spanning tree BPDU guard.

Example:

```
Switch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# bpdu-guard enable

Switch(easyport/ip-phone)#
```

13.2.4 *end*

Using this command you can finish IP Phone settings and return.

Syntax:

```
end
```

Example:

```
Switch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# end

Switch(easyport/ip-phone)#
```

13.2.5 *psec-action*

Using this command you can configure port security action for IP Phone.

Syntax:

```
psec-action      both|none|shutdown|trap
```

Parameter:

both

Send a SNMP trap and shutdown the port.

none

Do nothing.

shutdown

Shutdown the port.

trap

Send a SNMP trap.

Example:

```
Switch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# psec-action both

Switch(easyport/ip-phone)#
```

13.2.6 *psec-limit*

Using this command you can configure port security maximum for IP Phone.

Syntax:

```
psec-limit      <1-50>
```

Parameter:

<1-50>

Max. number of MAC addresses.

Example:

```
Switch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# psec-limit 40

Switch(easyport/ip-phone)#
```

13.2.7 *psec-mode*

Using this command you can configure port security mode for IP Phone.

Syntax:

```
psec-mode <1-50>
```

Parameter:**disable**

Disables port security.

enable

Enables port security.

Example:

```
Switch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# psec-mode enable

Switch(easyport/ip-phone)#
```

13.2.8 *show*

Using this command you can display current IP Phone settings.

Syntax:

```
show
```

Example:

```
Switch(easyport/ip-phone)# show
Role                : IP-PHONE
Access VLAN         : 8
VLAN Mode           : Trunk
Traffic Class       : 7
Port Security Mode   : Enabled
Port Security Action : Trap & Shutdown
Port Security Limit  : 40
STP Admin Edge      : Enabled
STP BPDU Guard      : Enabled
```

13.2.9 *traffic-class*

Using this command you can configure traffic class for IP Phone.

Syntax:

```
traffic-class <0-7>
```

Parameter:

<0-7>

0:Low, 7:High.

Example:

```
Switch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# traffic-class 7

Switch(easyport/ip-phone)#
```

13.2.10 *vlan-mode*

Using this command you can configure VLAN mode for IP Phone.

Syntax:

```
vlan-mode access|hybrid|trunk
```

Parameter:**access**

Untag all frames.

hybrid

Tag all frames except VLAN ID same as PVID.

trunk

Tag all frames.

Example:

```
Switch(easyport)# ip-phone 22
Switch(easyport/ip-phone)# vlan-mode trunk

Switch(easyport/ip-phone)#
```

13.3 *wifi-ap*

Using this command you can configure wireless access point parameters.

Syntax:

```
wifi-ap <port-list>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```

witch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# access-vlan 55
Switch(easyport/wifi-ap)# admin-edge disable
Switch(easyport/wifi-ap)# bpdu-guard disable
Switch(easyport/wifi-ap)# psec-action both
Switch(easyport/wifi-ap)# psec-limit 30
Switch(easyport/wifi-ap)# psec-mode enable
Switch(easyport/wifi-ap)# traffic-class 4
Switch(easyport/wifi-ap)# vlan-mode hybrid
Switch(easyport/wifi-ap)# show
Role                               : WIFI-AP
Access VLAN                       : 55
VLAN Mode                        : Hybrid
Traffic Class                     : 4
Port Security Mode                : Enabled
Port Security Action              : Trap & Shutdown
Port Security Limit               : 30
STP Admin Edge                   : Disabled
STP BPDU Guard                   : Disabled

Switch(easyport/wifi-ap)#

```

13.3.1 *access-vlan*

Using this command you can configure access VLAN for WiFi AP.

Syntax:

```
access-vlan          <1-4094>
```

Parameter:**<1-4094>**

Access VLAN ID, possible values from 1 to 4094..

Example:

```

Switch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# access-vlan 8

Switch(easyport/wifi-ap)#

```

13.3.2 *admin-edge*

Using this command you can configure spanning tree admin-edge for WiFi AP.

Syntax:

```
admin-edge          disable|enable
```

Parameter:**disable**

Disables spanning tree admin edge.

enable

Enables spanning tree admin edge.

Example:

```
Switch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# admin-edge enable

Switch(easyport/wifi-ap)#
```

13.3.3 *bpdu-guard*

Using this command you can configure spanning tree BPDU guard for WiFi AP.

Syntax:

```
bpdu-guard          disable|enable
```

Parameter:**disable**

Disables spanning tree BPDU guard.

enable

Enables spanning tree BPDU guard.

Example:

```
Switch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# bpdu-guard enable

Switch(easyport/wifi-ap)#
```

13.3.4 *end*

Using this command you can finish WiFi AP settings and return.

Syntax:

```
end
```

Example:

```
Switch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# end

Switch(easyport/wifi-ap)#
```

13.3.5 *psec-action*

Using this command you can configure port security action for WiFi AP.

Syntax:

```
psec-action          both|none|shutdown|trap
```

Parameter:**both**

Send a SNMP trap and shutdown the port.

none

Do nothing.

shutdown

Shutdown the port.

trap

Send a SNMP trap.

Example:

```
Switch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# psec-action both

Switch(easyport/wifi-ap)#
```

13.3.6 *psec-limit*

Using this command you can configure port security maximum for WiFi AP.

Syntax:

```
psec-limit          <1-50>
```

Parameter:**<1-50>**

Max. number of MAC addresses.

Example:

```
Switch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# psec-limit 40

Switch(easyport/wifi-ap)#
```

13.3.7 *psec-mode*

Using this command you can configure port security mode for WiFi AP.

Syntax:

```
psec-mode          <1-50>
```

Parameter:**disable**

Disables port security.

enable

Enables port security.

Example:

```
Switch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# psec-mode enable

Switch(easyport/wifi-ap)#
```

13.3.8 *show*

Using this command you can display current WiFi AP settings.

Syntax:

```
show
```

Example:

```
Switch(easyport/wifi-ap)# show
Role                : IP-PHONE
Access VLAN         : 8
VLAN Mode           : Trunk
Traffic Class       : 7
Port Security Mode  : Enabled
Port Security Action : Trap & Shutdown
Port Security Limit : 40
STP Admin Edge      : Enabled
STP BPDU Guard      : Enabled
```

13.3.9 *traffic-class*

Using this command you can configure traffic class for WiFi AP.

Syntax:

```
traffic-class <0-7>
```

Parameter:

<0-7>

0:Low, 7:High.

Example:

```
Switch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# traffic-class 7

Switch(easyport/wifi-ap)#
```

13.3.10 *vlan-mode*

Using this command you can configure VLAN mode for WiFi AP.

Syntax:

```
vlan-mode          access|hybrid|trunk
```

Parameter:**access**

Untag all frames.

hybrid

Tag all frames except VLAN ID same as PVID.

trunk

Tag all frames.

Example:

```
Switch(easyport)# wifi-ap 22
Switch(easyport/wifi-ap)# vlan-mode trunk
Switch(easyport/wifi-ap)#
```

14 EEE Commands

The section which allows the user to inspect and configure the current EEE port settings.

EEE is a power saving option that reduces the power usage when there is very low traffic utilization (or no traffic).

EEE works by powering down circuits when there is no traffic. When a port gets data to be transmitted all circuits are powered up. The time it takes to power up the circuits is named wakeup time. The default wakeup time is 17 us for 1Gbit links and 30 us for other link speeds. EEE devices must agree upon the value of the wakeup time in order to make sure that both the receiving and transmitting device has all circuits powered up when traffic is transmitted. The devices can exchange information about the devices wakeup time using the LLDP protocol.

For maximizing the power saving, the circuit isn't started at once transmit data are ready for a port, but is instead queued until 3000 bytes of data are ready to be transmitted. For not introducing a large delay in case that data less then 3000 bytes shall be transmitted, data are always transmitted after 48 us, giving a maximum latency of 48 us + the wakeup time.

If desired it is possible to minimize the latency for specific frames, by mapping the frames to a specific queue (done with QOS), and then mark the queue as an urgent queue. When an urgent queue gets data to be transmitted, the circuits will be powered up at once and the latency will be reduced to the wakeup time.

Command	Function
mode	Configure EEE mode
show	Show EEE information
urgent-queue	Configure EEE urgent queue

14.1 mode

Using this command you can configure EEE parameters.

Syntax:

```
mode <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables Energy Efficient Ethernet.

enable

Enables Energy Efficient Ethernet.

Example:

```
Switch(eee)# mode 2 enable
Switch(eee)# show
Port  Mode      Urgent Queues
```

```

-----
1   Disabled  none
2   Enabled   none
3   Disabled  none

```

14.2 *show*

Using this command you can show the current EEE configuration.

Syntax:

```
show
```

Example:

```

Switch(eee)# show
Port  Mode      Urgent Queues
-----
1     Disabled  none
2     Disabled  none
3     Disabled  none

```

14.3 *urgent-queue*

Using this command you can configure EEE urgent queue parameters.

Syntax:

```
urgent-queue <port-list> <queue-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<queue-list>

Queue list, format : 1,3-5.

disable

Queue will postpone the transmission until 3000 bytes are ready to be transmitted.

enable

Queues set will activate transmission of frames as soon as any data is available.

Example:

```

Switch(eee)# urgent-queue 1 4 enable
Switch(eee)# show
Port  Mode      Urgent Queues
-----

```

1	Disabled	4
2	Enabled	none
3	Disabled	none

15 Event Commands

The function is used to set an Alarm trap and get the Event log. The Trap Events Configuration function is used to enable the switch to send out the trap information while pre-defined trap events occurred.

Command	Function
group	Configure trap event severity level
show	Show trap event configuration

15.1 *group*

Using this command you can configure the trap event severity level.

Syntax:

```
group <group-name> <0-7>
```

Parameter:**<group-name>**

Trap event group name

- ACL
- ACL_Log
- Access_Mgmt
- Auth_Failed
- Cold_Start
- Config_Info
- Firmware_Upgrade
- Import_Export
- LACP
- Link_Status
- Login
- Logout
- Loop_Protect
- Mgmt_IP_Change
- Module_Change
- NAS
- Password_Change
- Poe_Auto_Check
- Port_Security
- VLAN
- Warm_Start

<0-7>

Severity level

- <0> Emergency: system is unusable
- <1> Alert: action must be taken immediately
- <2> Critical: critical conditions
- <3> Error: error conditions
- <4> Warning: warning conditions
- <5> Notice: normal but significant condition
- <6> Informational: informational messages
- <7> Debug: debug-level messages

Example:

```
Switch(event)# group acl 5
Switch(event)# show
Group Name                      Severity Level
-----
ACL                             Notice
ACL_Log                         Debug
Access_Mgmt                     Info
Auth_Failed                     Warning
Cold_Start                     Warning
Config_Info                     Info
Firmware_Upgrade                Info
Import_Export                   Info
LACP                            Info
Passwd_Change                   Info
Port_Security                   Info
Thermal_Protect                 Info
VLAN                            Info
Warm_Start                      Warning
Switch(event)#
```

15.2 show

Using this command you can show the current trap event severity configuration.

Syntax:

```
show
```

Example:

```
Switch(event)# show
Group Name                      Severity Level
-----
ACL                             Critical
ACL_Log                         Debug
Access_Mgmt                     Info
Auth_Failed                     Warning
Cold_Start                     Warning
Config_Info                     Info
Firmware_Upgrade                Info
Import_Export                   Info
Link_Status                     Warning
Login                           Info
Logout                          Info
```

Mgmt_IP_Change	Info
Module_Change	Notice
NAS	Info
Passwd_Change	Info
Port_Security	Info
Thermal_Protect	Info
VLAN	Info
Warm_Start	Warning
Switch(event)#	

16 Fdb Commands

Filtering Data Base (Fdb) configuration gathers many functions, including MAC table information, static MAC learning, which cannot be categorized to some function type.

MAC table

Switching of frames is based upon the DMAC address contained in the frame. The switch builds up a table that maps MAC addresses to switch ports for knowing which ports the frames should go to (based upon the DMAC address in the frame). This table contains both static and dynamic entries. The static entries are configured by the network administrator if the administrator wants to do a fixed mapping between the DMAC address and switch ports. The frames also contain a MAC address (SMAC address), which shows the MAC address of the equipment sending the frame. The SMAC address is used by the switch to automatically update the MAC table with these dynamic MAC addresses. Dynamic entries are removed from the MAC table if no frame with the corresponding SMAC address has been seen after a configurable age time

Command	Function
age-time	Configure aging time of MAC address
delete	Delete commands
flush	Flush out dynamic learned MAC address
learning	Configure learning mode of switch ports
static-mac	Configure static MAC address
show	Show MAC address table information

16.1 *age-time*

Using this command you can configure the age-time of MAC address.

Syntax:

```
age-time disable|<10-1000000>
```

Parameter:

disable

Disables automatic aging.

<10-1000000>

Aging time in seconds.

Example:

```
Switch(fdb)# age-time 1000
Switch(fdb)# show configuration
Automatic Aging : Enabled
Aging Time : 1000 seconds
Port Learning Mode
```

```

-----
1      Auto
2      Auto
3      Auto
4      Auto
.....
Switch(fdb)#

```

16.2 *delete*

Using this command you can delete a static MAC address entry.

Syntax:

```
delete static-mac <mac-address> <1-4094>
```

Parameter:

static mac

Deletes a static MAC entry.

<mac-address>

MAC address in the format 0a-1b-2c-3d-4e-5f.

<1-4094>

VLAN ID, available values from 1 to 4094.

Example:

```

Switch(fdb)# static-mac 00-1F-3B-6A-3B-11 3 22
Switch(fdb)# show static-mac
No      VID      MAC Address      Ports
-----
1       3       00-1f-3b-6a-3b-11 22
Total static MAC address : 1
Switch(fdb)# delete static-mac 00-1F-3B-6A-3B-11 3
Switch(fdb)# show static-mac
Total static MAC address : 0
Switch(fdb)#

```

16.3 *flush*

Using this command you can flush all MAC addresses which have been learned dynamically.

Syntax:

```
flush
```

Example:

```
Switch(fdb)# flush
Switch(fdb)#
```

16.4 *learning*

Using this command you can configure the MAC address learning mode.

Syntax:

```
learning <port-list> auto|disable|secure
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

auto

Learning is done automatically as soon as a frame with unknown SMAC is received.

disable

Disables MAC address learning mode.

secure

Only static MAC entries are learned, all other frames are dropped.

Example:

```
Switch(fdb)# learning 2 disable
Switch(fdb)# learning 4 secure
Switch(fdb)# show configuration
Automatic Aging : Enabled
Aging Time : 300 seconds
Port  Learning Mode
----  -
1      Auto
2      Disabled
3      Auto
4      Secure
.....
Switch(fdb)#
```

16.5 *static-mac*

Using this command you can configure a static MAC address entry.

Syntax:

```
static-mac <mac-address> <1-4094> <port-list>|block
```

Parameter:**<mac-address>**

MAC address in the format 0a-1b-2c-3d-4e-5f.

<1-4094>

VLAN ID, available values from 1 to 4094.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

block

Blocks the specific MAC address for all ports.

Example:

```
Switch(fdb)# static-mac 00-1F-3B-6A-3B-11 33 2
Switch(fdb)# show static-mac
No      VID      MAC Address      Ports
-----
1       33       00-1f-3b-6a-3b-11 2
Total static MAC address : 1
Switch(fdb)#
```

16.6 show

Using this command you can show the current MAC address table.

Syntax:

```
show configuration|static-mac
show mac-table [mac-address <mac-address>]
               [port <port-list>]
               [vid <1-4094>]
```

Parameter:**configuration**

Shows MAC address table configuration.

static-mac

Shows static MAC address.

mac-table

Shows MAC address table.

mac-address

Shows MAC address table for specific MAC address.

<mac-address>

MAC address in the format 0a-1b-2c-3d-4e-5f.

port

Shows MAC address table for specific ports.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

vid

Shows MAC address table for specific VLAN.

<1-4094>

VLAN ID, available values from 1 to 4094.

Example:

```
Switch(fdb)# static-mac 00-1F-3B-6A-3B-11 33 2
Switch(fdb)# show static-mac
No      VID    MAC Address      Ports
-----
1       33     00-1f-3b-6a-3b-11 2
Total static MAC address : 1
Switch(fdb)#
```

17 Firmware Commands

This section describes how to upgrade Firmware. The Switch can be enhanced with more value-added functions by installing firmware upgrades.

Command	Function
show	Show information about active and alternate firmware images
swap	Activate the alternate firmware image
upgrade	Upgrade system firmware

17.1 *show*

Using this command you can show the active and alternate firmware image version information.

Syntax:

```
show
```

Example:

```
Switch(firmware)# show
Active Image
-----
Image      : managed
Version    : GS-2326P (standalone) v2.50
Date       : 2014-01-17T14:33:34+08:00

Alternate Image
-----
Image      : managed.bk
Version    : GS-2326P (standalone) v1.65
Date       : 2013-01-14T14:11:35+08:00

Switch(firmware)#
```

17.2 *swap*

Using this command you can swap the active firmware image into alternate firmware image and vice versa.

Syntax:

```
swap
```

Example:

```
Switch(firmware)# swap
Alternate image activated, now rebooting.
```

```
Switch(firmware)#

Using username "admin".
admin@192.168.2.110's password:

Type 'help' or '?' to get help.
Switch# firmware
Switch(firmware)# show
Active Image
-----
Image       : managed
Version     : GS-2326P (standalone) v1.65
Date        : 2013-01-14T14:11:35+08:00

Alternate Image
-----
Image       : managed.bk
Version     : GS-2326P (standalone) v2.50
Date        : 2014-01-17T14:33:34+08:00

Switch(firmware)#
```

17.3 upgrade

Using this command you can upgrade the system firmware to active or alternate division.



While the firmware is being updated, the device access via web interface is disabled. The front LED flashes Green/Off with a frequency of 10 Hz while the firmware update is in progress. Do not restart or power off the device at this time or the switch may fail to work afterwards.

Syntax:

```
upgrade <ipv6-address> <filename>
upgrade <ip-hostname> <filename>
```

Parameter:

<ipv6-address>

TFTP server IPv6 address. IPv6 address is in 128-bit records represented as eight fields of up to four hexadecimal digits with a colon separate each field (":"). For example, **fe80: :215:c5ff: fe03:4dc7**. The symbol "::" is a special syntax that can be used as a shorthand way of representing multiple 16-bit groups of contiguous zeros; but it can only appear once. It also used a following legally IPv4 address. For example, **::192.1.2.34**.

<ip-hostname>

TFTP server IPv4 address or hostname.

<filename>

Firmware image file name

Example:

```
Switch(firmware)# upgrade 192.168.1.100 managed.bk
Switch(firmware)# show
Active Image
```

```
-----  
Image      : managed.bk  
Version    : GS-2326P (standalone) v1.65  
Date       : 2013-01-14T14:11:35+08:00
```

Alternate Image

```
-----  
Image      : managed  
Version    : GS-2326P (standalone) v2.50  
Date       : 2014-01-17T14:33:34+08:00
```

```
Switch(firmware)#
```

18 GARP Commands

The Generic Attribute Registration Protocol (GARP) provides a generic framework whereby devices in a bridged LAN, e.g. end stations and switches, can register and de-register attribute values, such as VLAN Identifiers, with each other. In doing so, the attributes are propagated to devices in the bridged LAN, and these devices form a reachability tree that is a subset of an active topology. GARP defines the architecture, rules of operation, state machines and variables for the registration and de-registration of attribute values.

A GARP participation in a switch or an end station consists of a GARP application component, and a GARP Information Declaration (GID) component associated with each port or the switch. The propagation of information between GARP participants for the same application in a bridge is carried out by the GARP Information Propagation (GIP) component. Protocol exchanges take place between GARP participants by means of LLC Type 1 services, using the group MAC address and PDU format defined for the GARP application concerned.

Command	Function
applicant	Enable/Disable applicant administrative control
join-time	Set the GARP join timer configuration
leave-all	Set the GARP leave all timer configuration
leave-time	Set the GARP leave timer configuration
show	Show the GARP configuration

18.1 *applicant*

Using this command you can enable or disable the applicant administrative control.

Syntax:

```
applicant <port-list> non-participant|normal-participant
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

non-participant

Set applicant administrative control to non-participant

normal-participant

Set applicant administrative control to normal-participant.

Example:

```
Switch(garp)# applicant 3 non-participant  
Switch(garp)#
```

18.2 *join-time*

Using this command you can set the GARP join time.

 If you didn't set the GARP environment already then the switch will show "Set jointimer failed".

Syntax:

```
join-time <port-list> <time-value>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<time-value>

Join time in seconds. Possible values from 200 to 1000.

Example:

```
Switch(garp)# join-time 3-5 200
Error! Set jointimer failed
```

18.3 *leave-all*

Using this command you can set the GARP leave all time.

 If you didn't set the GARP environment already then the switch will show "Set leave all timer failed".

Syntax:

```
leave-all <port-list> <timer-value>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<timer-value>

Leave all time in seconds. Possible values from 10000 to 50000.

Example:

```
Switch(garp)# leave-all 3-5 10000
Error! Set leavealltimer failed
Switch(garp)#
```

18.4 *leave-time*

Using this command you can set GARP leave time.



If you didn't set the GARP environment already then the switch will show "Set leavetimer failed".

Syntax:

```
leave-time <port-list> <timer-value>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<timer-value>

Leave all time in seconds. Possible values from 600 to 3000.

Example:

```
Switch(garp)# leave-time 3-5 600
Error! Set leavetimer failed
Switch(garp)#
```

18.5 *show*

Using this command you can display the current GARP configuration and statistics.



If you didn't set the GARP environment already then the switch will show "empty field value".

Syntax:

```
show config
show statistic <port-list>
```

Parameter:

config

Shows current GARP configuration.

statistic

Shows the basic GARP port statistics.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(garp)# show statistic 3-5 ?
<cr>
Switch(garp)# show statistic 3-5
Port  Peer MAC          Failed Count
-----
3      -                  -
4      -                  -
5      -                  -
Switch(garp)#
Switch(garp)#
```

19 GVRP Commands

GVRP is an application based on Generic Attribute Registration Protocol (GARP), mainly used to automatically and dynamically maintain the group membership information of the VLANs. The GVRP offers the function providing the VLAN registration service through a GARP application. It makes use of GARP Information Declaration (GID) to maintain the ports associated with their attribute database and GARP Information Propagation (GIP) to communicate among switches and end stations. With GID information and GIP, GVRP state machine maintain the contents of Dynamic VLAN Registration Entries for each VLAN and propagate these information to other GVRP-aware devices to setup and update their knowledge database, the set of VLANs associated with currently active members, and through which ports these members can be reached.

Command	Function
clear	Clear the basic GVRP port statistics
control	Enable/Disable GVRP globally
mode	Enable/Disable GVRP on port
rrole	Enable/Disable GVRP restricted role on port
show	Show the GVRP configuration

19.1 *clear*

Using this command you can clear the basic GVRP port statistics.



If you set the GVRP on port then you could show the port GVRP statistics information or clear all record on port.

Syntax:

```
clear <port-list>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(gvrp)# clear 3-5
Switch(gvrp)#
```

19.2 *control*

Using this command you can enable or disable the GVRP mode globally.

Syntax:

```
control disable|enable
```

Parameter:**disable**

Disables GVRP function globally.

enable

Enables GVRP function globally.

Example:

```
Switch(gvrp)# control enable
Switch(gvrp)#
```

19.3 *mode*

Using this command you can enable or disable the GVRP function per port.

Syntax:

```
mode <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables GVRP function on the selected ports.

enable

Enables GVRP function on the selected ports.

Example:

```
Switch(gvrp)# mode 3-5 enable
Switch(gvrp)#
```

19.4 *role*

Using this command you can enable or disable the GVRP restricted role per port.

Syntax:

```
rrole <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables GVRP restricted role on the selected ports.

enable

Enables GVRP restricted role on the selected ports.

Example:

```
Switch(gvrp)# rrole 3-5 enable
Switch(gvrp)#
```

19.5 show

Using this command you can display the current GVRP information.

Syntax:

```
show config
show statistics <port-list>
```

Parameter:**config**

Shows the current GVRP configuration.

statistics

Shows the basic GVRP port statistics.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(gvrp)# show config
GVRP global mode : Enabled

Port  Mode      Restricted Role
----  -
1     Disabled    Disabled
2     Disabled    Disabled
3     Enabled     Enabled
4     Enabled     Enabled
5     Enabled     Enabled
6     Disabled    Disabled
7     Disabled    Disabled
8     Disabled    Disabled
9     Disabled    Disabled
.....
```

```
Switch(gvrp)#  
Switch(gvrp)# show statistics 1-10  
Port   Joins Tx Count      Leaves Tx Count  
-----  
1       0                   0  
2       0                   0  
3       0                   0  
4       0                   0  
5       0                   0  
6       0                   0  
7       0                   0  
8       0                   0  
9       0                   0  
.....  
Switch(gvrp)#
```

20 HTTPS Commands

This section shows you how to use HTTPS to securely access the Switch. HTTPS is a secure communication protocol that combines authentication and data encryption to provide secure encrypted communication via the browser.

Command	Function
mode	Configure the HTTPS mode
redirect	Configure the HTTPS redirect mode
show	Show the HTTPS configuration

20.1 *mode*

Using this command you can enable or disable HTTPS operation.

Syntax:

```
mode disable|enable
```

Parameter:

disable

Disables HTTPS operation.

enable

Enables HTTPS operation.

Example:

```
Switch(https)# mode enable
Switch(https)#
```

20.2 *redirect*

Using this command you can enable or disable HTTPS redirect mode.

Syntax:

```
redirect disable|enable
```

Parameter:

disable

Disables redirect HTTPS mode.

enable

Enables redirect HTTPS mode.

Example:

```
Switch(https)# redirect enable
Switch(https)#
```

20.3 *show*

Using this command you can display the current HTTPS configuration.

Syntax:

show

Example:

```
Switch(https)# show
HTTPS Mode      : Enabled
HTTPS Redirect Mode : Enabled
Switch(https)#
```

21 IGMP Commands

The function, is used to establish the multicast groups to forward the multicast packet to the member ports, and, in nature, avoids wasting the bandwidth while IP multicast packets are running over the network. This is because a switch that does not support IGMP or IGMP Snooping cannot tell the multicast packet from the broadcast packet, so it can only treat them all as the broadcast packet. Without IGMP Snooping, the multicast packet forwarding function is plain and nothing is different from broadcast packet.

A switch supported IGMP Snooping with the functions of query, report and leave, a type of packet exchanged between IP Multicast Router/Switch and IP Multicast Host, can update the information of the Multicast table when a member (port) joins or leaves an IP Multicast Destination Address. With this function, once a switch receives an IP multicast packet, it will forward the packet to the members who joined in a specified IP multicast group before.

The packets will be discarded by the IGMP Snooping if the user transmits multicast packets to the multicast group that had not been built up in advance. IGMP mode enables the switch to issue IGMP function that you enable IGMP proxy or snooping on the switch, which connects to a router closer to the root of the tree. This interface is the upstream interface. The router on the upstream interface should be running IGMP.

Command	Function
compatibility	Set the Versions of IGMP Operating on Hosts and Routers
delete	Delete commands what you set on the switch
fast-leave	Set per-port Fast Leave
filtering	The IP Multicast Group that will be filtered
flooding	Set IGMP Flooding Mode
lmqi	Set per-VLAN Last Member Query Interval
proxy	Set IGMP Proxy Mode
qi	Set per-VLAN Query Interval
qri	Set per-VLAN Query Response Interval
querier	Set per-VLAN IGMP Querier
router	Set Router Port
rv	Set per-VLAN Robustness Variable
show	Show IGMP Snooping Information
snooping	Set IGMP Snooping Mode
ssm-range	Set IGMP SSM Range
state	Enable/Disable per-VLAN IGMP Snooping Mode
throttling	Set per-port Throttling
uri	Set per-VLAN Unsolicited Report Interval

21.1 *compatibility*

Using this command you can configure the IGMP compatibility.

Syntax:

```
compatibility <vlan-list> Forced-IGMPv1|Forced-IGMPv2|Forced-IGMPv3|IGMP-Auto
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

Forced-IGMPv1

Enables IGMPv1 of IGMP operating on hosts and routers.

Forced-IGMPv2

Enables IGMPv2 of IGMP operating on hosts and routers.

Forced-IGMPv3

Enables IGMPv3 of IGMP operating on hosts and routers.

IGMP-Auto

Enables auto mode of IGMP operating on hosts and routers.

Example:

```
Switch(igmp)# compatibility 1 IGMP-Auto
Switch(igmp)# show status 1
  Querier Rx      Tx      Rx      Rx      Rx      Rx
VID  Status Query   Query   V1 Join  V2 Join  V3 Join  V2
Leave
-----
Switch(igmp)#
```

21.2 delete

Using this command you can delete IGMP filtering groups.



If you type illegal ipmc-address, then switch won't allow you to delete it. And screen will display e.g. "Invalid argument '223.224.223.224'".

Syntax:

```
delete <port-list> <ipmc-address>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<ipmc-address>

IPMC address to delete IGMP filtering group. Available range from 224.0.0.0 to 239.255.255.255.

Example:

```
Switch(igmp)# delete 3 224.0.0.2
Switch(igmp)#
```

21.3 *fast-leave*

Using this command you can configure the fast-leave mode for each port.

 When you enable IGMP fast-leave processing, the switch immediately removes a port when it detects an IGMP version 2 leave message on that port.

Syntax:

```
fast-leave <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables fast-leave mode for the selected ports.

enable

Enables fast-leave mode for the selected ports.

Example:

```
Switch(igmp)# fast-leave 1 disable
Switch(igmp)#
```

21.4 *filtering*

Using this command you can configure IGMP filtering group for each port.

 If you type illegal ipmc-address, then switch won't allow you to filter it. And screen will display e.g. "Invalid argument '223.224.223.224'".

Syntax:

```
filtering <port-list> <ipmc-address>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<ipmc-address>

IPMC address of the IGMP filtering group. Available range from 224.0.0.0 to 239.255.255.255.

Example:

```
Switch(igmp)# filtering 5 224.0.0.1
Switch(igmp)#
```

21.5 *flooding*

Using this command you can enable or disable the IGMP flooding mode.

Syntax:

```
flooding enable|disable
```

Parameter:**disable**

Disables the IGMP flooding mode.

enable

Enables the IGMP flooding mode.

Example:

```
Switch(igmp)# flooding enable
Switch(igmp)# show config
IGMP Snooping : Disabled

IGMP Flooding Control : Enabled
IGMP Proxy : Disabled

IGMP SSM Range: 232.0.0.0/8
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
-----
1     Disabled    No              Disabled    Unlimited
2     Disabled    No              Disabled    Unlimited
3     Disabled    No              Disabled    Unlimited
4     Disabled    No              Disabled    Unlimited
5     Disabled    No              Disabled    Unlimited
6     Disabled    No              Disabled    Unlimited
7     Disabled    No              Disabled    Unlimited
8     Disabled    No              Disabled    Unlimited
9     Disabled    No              Disabled    Unlimited
.....
Switch(igmp)#
```

21.6 *lmqi*

Using this command you can define Last Member Query Interval for each VLAN.

Syntax:

```
lmqi <vlan-list> <0-31744>
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<0-31744>

Range:0~31744 tenths of sec, Default:100 tenths of sec

Example:

```
Switch(igmp)# lmqi 45 379
Switch(igmp)#
```

21.7 proxy

Using this command you can enable or disable the IGMP proxy mode.

Syntax:

```
proxy enable|disable
```

Parameter:**enable**

Enables IGMP proxy mode.

disable

Disables IGMP proxy mode.

Example:

```
Switch(igmp)# proxy enable
Switch(igmp)# show config

IGMP Snooping : Disabled

IGMP Flooding Control : Enabled
IGMP Proxy : Enabled

IGMP SSM Range: 232.0.0.0/8
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
-----
1      Disabled      No              Disabled     Unlimited
2      Disabled      No              Disabled     Unlimited
3      Disabled      No              Disabled     Unlimited
4      Disabled      No              Disabled     Unlimited
5      Disabled      No              Disabled     Unlimited
6      Disabled      No              Disabled     Unlimited
7      Disabled      No              Disabled     Unlimited
8      Disabled      No              Disabled     Unlimited
9      Disabled      No              Disabled     Unlimited
```

```
.....  
Switch(igmp)#
```

21.8 *qi*

Using this command you can define Query Interval for each VLAN.

Syntax:

```
qi <vlan-list> <1-31744>
```

Parameter:

<vlan-list>

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<1-31744>

Range:1~31744 sec, default:125 sec

Example:

```
Switch(igmp)# qi 5 360  
Switch(igmp)#
```

21.9 *qri*

Using this command you can define Query Response Intervall for each VLAN.

Syntax:

```
qri <vlan-list> <0-31744>
```

Parameter:

<vlan-list>

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<0-31744>

Range:0~31744 tenths of sec, default:100 tenths of sec

Example:

```
Switch(igmp)# qri 5 360  
Switch(igmp)#
```

21.10 *querier*

Using this command you can enable or disable IGMP Querier mode for each VLAN.

Syntax:

```
querier <vlan-list> disable|enable
```

Parameter:

<vlan-list>

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

disable

Disables IGMP Querier mode for the selected VLAN.

enable

Enables IGMP Querier mode for the selected VLAN.

Example:

```
Switch(igmp)# querier 5 enable
Switch(igmp)#
```

21.11 *router*

Using this command you can enable or disable IGMP Router Port mode for each port.

Syntax:

```
router <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables IGMP Router Port mode for each port.

enable

Enables IGMP Router Port mode for each port.

Example:

```
Switch(igmp)# router 5 enable
Switch(igmp)#
```

21.12 *rv*

Using this command you can define Robustness Variable for each VLAN.

Syntax:

```
rv <vlan-list> <1-255>
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<1-255>

Range:1~255, default:2

Example:

```
Switch(igmp)# rv 5 100
Switch(igmp)#
```

21.13 *show*

Using this command you can display the IGMP Snooping information.

Syntax:

```
show config
show groups|ssm|status|version [<1-4094>]
```

Parameter:**config**

Shows IGMP snooping configuration.

groups

Shows IGMP group table.

ssm

Shows IGMPv3 information table.

status

Shows IGMP snooping status.

version

Shows current IGMP working querier/host version.

<1-4094>

VLAN ID, possible values from 1 to 4094.

Example:

```
Switch(igmp)# show config
IGMP Snooping : Disabled

IGMP Flooding Control : Disabled
IGMP Proxy : Disabled

IGMP SSM Range: 232.0.0.0/8
Port Router Dynamic Router Fast Leave Group Throttling Number
-----
1 Disabled No Disabled Unlimited
2 Disabled No Disabled Unlimited
3 Disabled No Disabled Unlimited
4 Disabled No Disabled Unlimited
5 Disabled No Disabled Unlimited
6 Disabled No Disabled Unlimited
7 Disabled No Disabled Unlimited
8 Disabled No Disabled Unlimited
9 Disabled No Disabled Unlimited
10 Disabled No Disabled Unlimited
11 Disabled No Disabled Unlimited
12 Disabled No Disabled Unlimited
13 Disabled No Disabled Unlimited
14 Disabled No Disabled Unlimited
15 Disabled No Disabled Unlimited
--More--, q to quit
```

21.14 *snooping*

Using this command you can enable or disable IGMP Snooping mode.

Syntax:

```
snooping disable|enable
```

Parameter:**disable**

Disables the global IGMP snooping mode.

enable

Enables the global IGMP snooping mode.

Example:

```
Switch(igmp)# snooping enable
Switch(igmp)#
```

21.15 *ssm-range*

Using this command you can define the IGMP SSM range.

Syntax:

```
ssm-range <ipmc-address> <4-32>
```

Parameter:**<ipmc-address>**

IGMP SSM range address.

<4-32>

IGMP SSM range value.

Example:

```
Switch(igmp)# ssm-range 224.0.0.1 16
Switch(igmp)#
```

21.16 *state*

Using this command you can enable or disable IGMP Snooping mode for each VLAN.

Syntax:

```
state <vlan-list> disable|enable
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

disable

Disables IGMP snooping for selected VLAN.

enable

Enables IGMP snooping for selected VLAN.

Example:

```
Switch(igmp)# state 5 enable
Switch(igmp)#
```

21.17 *throttling*

Using this command you can define the throttling value for each port.

Syntax:

```
throttling <port-list> <0-10>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-10>

Port group limit number, range:0~10, 0: unlimited

Example:

```
Switch(igmp)# throttling 5 8
Switch(igmp)#
```

21.18 *uri*

Using this command you can define Unsolicited Report Interval for each VLAN.

Syntax:

```
uri <vlan-list> <0-31744>
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<0-31744>

Range:0~31744 sec, default:1 sec

Example:

```
Switch(igmp)# uri 5 3600
Switch(igmp)#
```

22 IP Commands

IP is an acronym for Internet Protocol. It is a protocol used for communicating data across an internet network.

IP is a "best effort" system, which means that no packet of information sent over is assured to reach its destination in the same condition it was sent. Each device connected to a Local Area Network (LAN) or Wide Area Network (WAN) is given an Internet Protocol address, and this IP address is used to identify the device uniquely among all other devices connected to the extended network.

The current version of the Internet protocol is IPv4, which has 32-bits Internet Protocol addresses allowing for in excess of four billion unique addresses. This number is reduced drastically by the practice of webmasters taking addresses in large blocks, the bulk of which remain unused. There is a rather substantial movement to adopt a new version of the Internet Protocol, IPv6, which would have 128-bits Internet Protocol addresses. This number can be represented roughly by a three with thirty-nine zeroes after it. However, IPv4 is still the protocol of choice for most of the Internet.

Command	Function
dhcp	Enable/Disable DHCP client
dns-proxy	Enable/Disable DNS proxy
mgmt-vlan	Set the management VLAN ID
name-server	Set DNS IP address
setup	Set the IP address
show	Show ip information

22.1 *dhcp*

Using this command you can configure the DHCP client.

Syntax:

```
dhcp disable|enable|renew
```

Parameter:

disable

Disables DHCP client.

enable

Enables DHCP client.

renew

Forces DHCP client to renew IP address.

Example:

```
Switch(ip)# dhcp enable
Switch(ip)# show
                Configured      Current
                -----
DHCP Client    : Enabled
```

```

IP Address      : 192.168.20.1    0.0.0.0
IP Mask         : 255.255.255.0   0.0.0.0
IP Gateway      : 192.168.20.250  0.0.0.0
VLAN ID         : 1              1
DNS Server      : 0.0.0.0         0.0.0.0
DNS Proxy       : Enabled

```

22.2 *dns-proxy*

Using this command you can configure the DNS proxy.

Syntax:

```
dns-proxy disable|enable
```

Parameter:

disable

Disables DNS proxy mode.

enable

Enables DNS proxy mode.

Example:

```

Switch(ip)# dns-proxy enable
Switch(ip)# show

```

	Configured	Current
	-----	-----
DHCP Client	: Enabled	
IP Address	: 192.168.20.1	0.0.0.0
IP Mask	: 255.255.255.0	0.0.0.0
IP Gateway	: 192.168.20.250	0.0.0.0
VLAN ID	: 1	1
DNS Server	: 0.0.0.0	0.0.0.0
DNS Proxy	: Enabled	

22.3 *mgmt-vlan*

Using this command you can define the management VLAN ID.

Syntax:

```
mgmt-vlan <1-4094>
```

Parameter:

<1-4094>

Management VLAN ID, possible values from 1 to 4094

Example:

```
Switch(ip)# mgmt-vlan 2
Switch(ip)# show
```

	Configured	Current
	-----	-----
DHCP Client	: Disabled	
IP Address	: 192.168.20.1	192.168.20.1
IP Mask	: 255.255.255.0	255.255.255.0
IP Gateway	: 192.168.20.250	192.168.20.250
VLAN ID	: 2	2
DNS Server	: 0.0.0.0	0.0.0.0
DNS Proxy	: Disabled	

22.4 name-server

Using this command you can define the DNS IP address.

Syntax:

```
name-server <ip-address>
```

Parameter:

<ip-address>

DNS IP address.

Example:

```
Switch(ip)# name-server 192.168.20.10
Switch(ip)# show
```

	Configured	Current
	-----	-----
DHCP Client	: Disabled	
IP Address	: 192.168.20.1	192.168.20.1
IP Mask	: 255.255.255.0	255.255.255.0
IP Gateway	: 192.168.20.250	192.168.20.250
VLAN ID	: 2	2
DNS Server	: 192.168.20.10	192.168.20.10
DNS Proxy	: Disabled	

22.5 setup

Using this command you can define the devices IP address.



The IP address and the router must be on the same subnet.

Syntax:

```
setup <ip-address> [<ip-mask>] [<ip-address>]
```

Parameter:**<ip-address>**

IP address.

<ip-mask>

IP subnet mask.

<ip-address>

Gateway IP address.

Example:

```
Switch(ip)# setup 192.168.20.10 255.255.255.0 192.168.20.250
Switch(ip)# show
```

	Configured	Current
	-----	-----
DHCP Client	: Disabled	
IP Address	: 192.168.20.10	192.168.20.10
IP Mask	: 255.255.255.0	255.255.255.0
IP Gateway	: 192.168.20.250	192.168.20.250
VLAN ID	: 2	2
DNS Server	: 0.0.0.0	0.0.0.0
DNS Proxy	: Disabled	

22.6 show

Using this command you can show the current IP information.

Syntax:*show***Example:**

```
Switch(ip)# show
```

	Configured	Current
	-----	-----
DHCP Client	: Disabled	
IP Address	: 192.168.20.10	192.168.20.10
IP Mask	: 255.255.255.0	255.255.255.0
IP Gateway	: 192.168.20.250	192.168.20.250
VLAN ID	: 2	2
DNS Server	: 0.0.0.0	0.0.0.0
DNS Proxy	: Disabled	

23 IP-Source-Guard Commands

The section describes to configure the IP Source Guard detail parameters of the switch. You could use the IP Source Guard configure to enable or disable with the Port of the switch.

Command	Function
add	Add or modify IP source guard static entry
delete	Delete IP source guard static entry
limit	IP source guard port limitation for dynamic entries
mode	Configure IP source guard mode
port-mode	Configure IP source guard port mode
show	Show IP source guard information
translate	Translate IP source guard dynamic entries into static entries

23.1 *add*

Using this command you can add or modify IP source guard static entry.

Syntax:

```
add <port-list> <1-4094> <ip-address> <mac-address>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<1-4094>

VLAN ID, possible values from 1 to 4094.

<ip-address>

IP address allowed for doing IP source guard.

<mac-address>

MAC address, format 0a-1b-2c-3d-4e-5f.

Example:

```
Switch(ip-source-guard)# add 5 2 192.168.2.100 0a-1b-2c-3d-4e-5f
Switch(ip-source-guard)# show binding-table
Type      Port  VLAN  IP Address      MAC Address
-----
Static    5     2     192.168.2.100  0a-1b-2c-3d-4e-5f
```

23.2 delete

Using this command you can delete statis IP source guard entries.

Syntax:

```
delete <port-list> <1-4094> <ip-address> <mac-address>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<1-4094>

VLAN ID, possible values from 1 to 4094.

<ip-address>

IP address.

<mac-address>

MAC address, format 0a-1b-2c-3d-4e-5f.

Example:

```
Switch(ip-source-guard)# delete 5 2 192.168.2.100 0a-1b-2c-3d-4e-5f
Switch(ip-source-guard)# show binding-table
Type      Port  VLAN  IP Address      MAC Address
-----
<none>
```

23.3 limit

Using this command you can define IP source guard port limitation for dynamic entries.

Syntax:

```
limit <port-list> <0-2>|unlimited
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-2>

Specify the maximum number of dynamic clients that can be learned on given port. If the port mode is enabled and the value of max dynamic client is equal to 0, it means only allow the IP packets forwarding that are matched in static entries on the specific port unlimited

unlimited

dynamic clients

Example:

```
Switch(ip-source-guard)# limit 1 0
Switch(ip-source-guard)# show config
```

IP Source Guard Mode : Disabled

Port	Port Mode	Dynamic Entry Limit
1	Disabled	0
2	Disabled	unlimited
3	Disabled	unlimited
4	Disabled	unlimited
5	Disabled	unlimited
6	Disabled	unlimited
7	Disabled	unlimited
8	Disabled	unlimited
9	Disabled	unlimited
10	Disabled	unlimited
11	Disabled	unlimited
12	Disabled	unlimited
13	Disabled	unlimited
14	Disabled	unlimited
15	Disabled	unlimited
16	Disabled	unlimited
17	Disabled	unlimited
18	Disabled	unlimited
19	Disabled	unlimited
20	Disabled	unlimited
21	Disabled	unlimited
22	Disabled	unlimited
23	Disabled	unlimited
24	Disabled	unlimited
25	Disabled	unlimited
26	Disabled	unlimited
27	Disabled	unlimited
28	Disabled	unlimited

23.4 mode

Using this command you can enable or disable the IP source guard mode.

Syntax:

```
mode enable|disable
```

Parameter:**disable**

Globally disable IP source guard mode.

enable

Globally enable IP source guard mode. All configured ACEs will be lost when the mode is enabled.

Example:

```
Switch(ip-source-guard)# mode enable
Switch(ip-source-guard)# show config

IP Source Guard Mode : Enabled

Port  Port Mode  Dynamic Entry Limit
-----
1      Disabled  0
2      Disabled  unlimited
3      Disabled  unlimited
4      Disabled  unlimited
5      Disabled  unlimited
6      Disabled  unlimited
7      Disabled  unlimited
8      Disabled  unlimited
9      Disabled  unlimited
10     Disabled  unlimited
11     Disabled  unlimited
12     Disabled  unlimited
13     Disabled  unlimited
14     Disabled  unlimited
15     Disabled  unlimited
16     Disabled  unlimited
17     Disabled  unlimited
18     Disabled  unlimited
19     Disabled  unlimited
20     Disabled  unlimited
21     Disabled  unlimited
22     Disabled  unlimited
23     Disabled  unlimited
24     Disabled  unlimited
25     Disabled  unlimited
26     Disabled  unlimited
27     Disabled  unlimited
28     Disabled  unlimited
```

23.5 *port-mode*

Using this command you can enable or disable IP source guard port mode for each port.

Syntax:

```
port-mode <port-list> enable|disable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables IP source guard port mode.

enable

Enables IP source guard port mode.

Example:

```
Switch(ip-source-guard)# port-mode 1 enable
Switch(ip-source-guard)# show config
```

```
IP Source Guard Mode : Enabled
```

Port	Port Mode	Dynamic Entry Limit
1	Enabled	unlimited
2	Disabled	unlimited
3	Disabled	unlimited
4	Disabled	unlimited
5	Disabled	unlimited
6	Disabled	unlimited
7	Disabled	unlimited
8	Disabled	unlimited
9	Disabled	unlimited
10	Disabled	unlimited
11	Disabled	unlimited
12	Disabled	unlimited
13	Disabled	unlimited
14	Disabled	unlimited
15	Disabled	unlimited
16	Disabled	unlimited
17	Disabled	unlimited
18	Disabled	unlimited
19	Disabled	unlimited
20	Disabled	unlimited
21	Disabled	unlimited
22	Disabled	unlimited
23	Disabled	unlimited
24	Disabled	unlimited
25	Disabled	unlimited
26	Disabled	unlimited
27	Disabled	unlimited
28	Disabled	unlimited

23.6 show

Using this command you can show the current IP source guard information.

Syntax:

```
show binding-table|config
```

Parameter:**binding-table**

Shows IP-MAC binding table.

config

Shows IP source guard configuration.

Example:

```
Switch(ip-source-guard)# show binding-table
Type      Port  VLAN  IP Address      MAC Address
-----
Static    1     1     192.168.1.1     5a-80-70-64-60-80
```

23.7 *translate*

Using this command you can translate dynamic IP source guard entries into static entries.

Syntax:

translate

Example:

```
Switch(ip-source-guard)# translate
IP Source Guard:
    Translate 0 dynamic entries into static entries.
```

24 IPv6 Commands

This section describes how to configure the switch-managed IPv6 information. The Configured column is used to view or change the IPv6 configuration. And the Current column is used to show the active IPv6 configuration.

Command	Function
autoconfig	Configure IPv6 autoconfig mode
setup	Set the IPv6 address
show	Show IPv6 information

24.1 *autoconfig*

Using this command you can configure the IPv6 autoconfig mode.

Syntax:

```
autoconfig disable|enable|renew
```

Parameter:

disable

Disables IPv6 autoconfig mode.

enable

Enables IPv6 autoconfig mode.

renew

Forces renewal of IPv6 address.

Example:

```
Switch(ipv6)# autoconfig enable
Switch(ipv6)# show config
Auto Configuration : Enabled
Address           : ::192.168.1.1
Prefix            : 96
Gateway           : ::
```

24.2 *setup*

Using this command you can define the devices IPv6 address.

Syntax:

```
setup setup <ipv6-address> [<1-128>] [<ipv6-gateway>]
```

Parameter:**<ipv6-address>**

IPv6 address is in 128-bit records represented as eight fields of up to four hexadecimal digits with a colon separate each field (":"). For example, **fe80::215:c5ff:fe03:4dc7**. The symbol "::" is a special syntax that can be used as a shorthand way of representing multiple 16-bit groups of contiguous zeros; but it can only appear once. It also used a following legally IPv4 address. For example, **::192.1.2.34**.

<1-128>

IPv6 prefix

<ipv6-gateway>

Gateway IPv6 address IPv6 address is in 128-bit records represented as eight fields of up to four hexadecimal digits with a colon separate each field (":").

Example:

```
Switch(ipv6)# setup ::192.168.6.1 1 ::192.168.0.0
Switch(ipv6)# show config
Auto Configuration : Enabled
Address           : ::192.168.6.1
Prefix            : 1
Gateway           : ::192.168.0.0
```

24.3 show

Using this command you can show the current IPv6 information.

Syntax:

```
show config|current
```

Parameter:**config**

Shows IPv6 configuration.

current

Shows current IPv6 information.

Example:

```
Switch(ipv6)# show config
Auto Configuration : Disabled
Address           : ::192.168.6.1
Prefix            : 96
Gateway           : ::

Switch(ipv6)# show current

Active Configuration for IPv6: (Static with Stateless)
Link-Local Address : fe80::240:c7ff:fe34:3400
Address            : ::192.168.6.1
```

24 IPv6 Commands

Prefix	:	96
Gateway	:	::

25 LACP Commands

Ports using Link Aggregation Control Protocol (according to IEEE 802.3ad specification) as their trunking method can choose their unique LACP GroupID to form a logic "trunked port". The benefit of using LACP is that a port makes an agreement with its peer port before it becomes a ready member of a "trunk group" (also called aggregator). LACP is safer than the other trunking method - static trunk.

Command	Function
clear	Clear command
key	Configure the LACP key
mode	Configure the LACP mode
role	Configure the LACP role
Show	Show LACP information

25.1 *clear*

Using this command you can clear the link aggregation entries.

Syntax:

```
clear statistics
```

Parameter:

statistics

Clears LACP statistics.

Example:

```
Switch(lacp)# clear statistics
Switch(lacp)# show statistics
Port  Rx Frames  Tx Frames  Rx Unknown  Rx Illegal
-----
1      0           0           0           0
2      0           0           0           0
3      0           0           0           0
4      0           0           0           0
```

25.2 *key*

Using this command you can configure the LACP key for each port.

Syntax:

```
key <port-list> <1-65535>|auto
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<1-65535>

LACP key

auto

The Auto setting will set the key as appropriate by the physical link speed, 10Mb = 1, 100Mb = 2, 1Gb = 3

Example:

```
Switch(lacp)# key 1 10000
Switch(lacp)# show config
Port  Mode      Key  Role
-----
1     Disabled  10000 Active
2     Disabled  Auto  Active
3     Disabled  Auto  Active
4     Disabled  Auto  Active
5     Disabled  Auto  Active
```

25.3 mode

Using this command you can configure the LACP mode for each port.

Syntax:

```
mode <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables LACP protocol for the selected port(s).

enable

Enables LACP protocol for the selected port(s).

Example:

```
Switch(lacp)# mode 1 enable
Switch(lacp)# show config
Port  Mode      Key  Role
-----
1     Enabled   Auto  Active
2     Disabled  Auto  Active
3     Disabled  Auto  Active
4     Disabled  Auto  Active
```

25.4 *role*

Using this command you can configure the LACP role for each port.

Syntax:

```
role <port-list> active|passive
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

active

Initiate LACP negotiation, and transmit LACP packets each second.

passive

Listen for LACP packets.

Example:

```
Switch(lacp)# role 1 passive
Switch(lacp)# show config
Port  Mode      Key      Role
----  -
1     Disabled  Auto     Passive
2     Disabled  Auto     Active
3     Disabled  Auto     Active
```

25.5 *show*

Using this command you can show LACP information.

Syntax:

```
show config|statistics|status
```

Parameter:

config

Shows LACP configuration.

statistics

Shows LACP statistics.

status

Shows LACP status.

Example:

```
Switch(lacp)# show config
Port  Mode      Key  Role
-----
1     Disabled  Auto  Passive
2     Disabled  Auto  Active

Switch(lacp)# show statistics
Port  Rx Frames  Tx Frames  Rx Unknown  Rx Illegal
-----
1     0          0          0           0
2     0          0          0           0
3     0          0          0           0

Switch(lacp)# show status

Port  Mode      Key  Aggr ID  Partner System ID  Partner Port
-----
1     Disabled  -    -        -                  -
2     Disabled  -    -        -                  -
3     Disabled  -    -        -                  -
```

26 LLDP Commands

The switch supports the LLDP. For current information on your switch model, The Link Layer Discovery Protocol (LLDP) provides a standards-based method for enabling switches to advertise themselves to adjacent devices and to learn about adjacent LLDP devices. The Link Layer Discovery Protocol (LLDP) is a vendor-neutral Link Layer protocol in the Internet Protocol Suite used by network devices for advertising their identity, capabilities, and neighbors on a IEEE 802 local area network, principally wired Ethernet. The protocol is formally referred to by the IEEE as Station and Media Access Control Connectivity Discovery specified in standards document IEEE 802.1AB.

Command	Function
<code>cdp-aware</code>	Configure CDP (Cisco Discovery Protocol) aware mode
<code>clear</code>	Clear LLDP statistics
<code>delay</code>	Configure ARP inspection mode
<code>hold</code>	Configure LLDP Tx hold value
<code>interval</code>	Configure LLDP transmission interval
<code>mode</code>	Configure the LLDP mode
<code>option-tlv</code>	Configure LLDP Optional TLVs
<code>reinit</code>	Configure LLDP reinit delay
<code>show</code>	Show LLDP information

26.1 *cdp-aware*

Using this command you can configure CDP (Cisco Discovery Protocol) aware mode.

Syntax:

```
cdp-aware <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables CDP awareness

enable

Enables CDP awareness (CDP discovery information is added to the LLDP neighbor table)

Example:

```
Switch(lldp)# cdp-aware 1 enable
Switch(lldp)# show config
Interval      : 30
Hold          : 4
```

```

Tx Delay      : 2
Reinit Delay: 2
Port          System      System      System      Management CDP
Mode          Description Name      Description Capability Address
awareness
-----
1  Disabled Enabled      Enabled Enabled      Enabled      Enabled      Enabled
2  Disabled Enabled      Enabled Enabled      Enabled      Enabled
Disabled
3  Disabled Enabled      Enabled Enabled      Enabled      Enabled
Disabled

```

26.2 *clear*

Using this command you can clear LLDP statistics.

Syntax:

```
clear
```

Example:

```

Switch(lldp)# clear
Switch(lldp)# show statistics
LLDP global counters
Neighbor entries was last changed at 2014-01-01 00:00:00 (5600 sec. ago).
Total Neighbors Entries Added 0.
Total Neighbors Entries Deleted 0.
Total Neighbors Entries Dropped 0.
Total Neighbors Entries Aged Out 0.

LLDP local counters

```

Port	Rx Frames	Tx Frames	Rx Errors	Rx Discards	Rx TLV Errors	Rx TLV Unknown	Rx TLV Organz.	Aged
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0

26.3 *delay*

Using this command you can configure LLDP Tx delay.

Syntax:

```
delay <1-8192>
```

Parameter:

<1-8192>

LLDP transmission delay

Example:

```
Switch(lldp)# delay 5
Switch(lldp)# show config
Interval      : 30
Hold          : 4
Tx Delay      : 5
Reinit Delay: 2
```

Port	Mode	Port Description	System Name	System Description	System Capability	Management Address	CDP awareness
1	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
2	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
3	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled

26.4 hold

Using this command you can configure LLDP Tx hold value.

Syntax:

```
hold <2-10>
```

Parameter:

<2-10>

LLDP hold value

Example:

```
Switch(lldp)# hold 10
Switch(lldp)# show config
Interval      : 30
Hold          : 10
Tx Delay      : 2
Reinit Delay: 2
```

Port	Mode	Port Description	System Name	System Description	System Capability	Management Address	CDP awareness
1	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
2	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
3	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled

26.5 *interval*

Using this command you can configure LLDP transmission interval.

Syntax:

```
interval <5-32768>
```

Parameter:

<5-32768>

LLDP transmission interval

Example:

```
Switch(lldp)# interval 40
Switch(lldp)# show config
Interval      : 40
Hold          : 4
Tx Delay      : 2
Reinit Delay: 2
```

Port	Mode	System Description	System Name	System Description	System Capability	Management Address	CDP awareness
1	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
2	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
3	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled

26.6 *mode*

Using this command you can configure the LLDP mode.

Syntax:

```
mode <port-list> disable|enable|rx-only|tx-only
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

The switch will not send out LLDP information, and will drop LLDP information received from neighbours.

enable

The switch will send out LLDP information, and will analyze LLDP information received from neighbours.

rx-only

The switch will not send out LLDP information, but LLDP information from neighbour units is analyzed.

tx-only

The switch will drop LLDP information received from neighbours, but will send out LLDP information.

Example:

```
Switch(lldp)# mode 1 enable
Switch(lldp)# show config
Interval      : 30
Hold          : 4
Tx Delay      : 2
Reinit Delay  : 2
```

Port	Mode	Port Description	System Name	System Description	System Capability	Management Address	CDP awareness
1	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
2	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
3	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled

26.7 option-tiv

Using this command you can configure LLDP Optional TLVs.

Syntax:

```
option-tiv <port-list> (mgmt-addr|port-desc|sys-cap|sys-desc|sys-name) (disable|enable)
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

mgmt-addr

Management IP address

port-desc

Port description

sys-cap

System capability

sys-desc

System description

sys-name

System name

disable

Disables TLV

enable

Enables TLV

Example:

```
Switch(lldp)# option-tlv 1 mgmt-addr disable
Switch(lldp)# option-tlv 1 port-desc disable
Switch(lldp)# option-tlv 1 sys-capability disable
Switch(lldp)# option-tlv 1 sys-desc disable
Switch(lldp)# option-tlv 1 sys-name disable
Switch(lldp)# show config
Interval      : 30
Hold          : 4
Tx Delay      : 2
Reinit Delay: 2
```

Port	Mode	Port Description	System Name	System Description	System Capability	Management Address	CDP Awareness
1	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled
2	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled
3	Disabled	Enabled	Enabled	Enabled	Enabled	Enabled	Enabled

26.8 *reinit*

Using this command you can configure LLDP reinit delay.

Syntax:*reinit* <1-10>**Parameter:**

<1-10>

LLDP reinit delay

Example:

```
Switch(lldp)# reinit 10
Switch(lldp)# show config
Interval      : 30
Hold          : 4
Tx Delay      : 2
Reinit Delay: 10
```

26.9 show

Using this command you can show current LLDP information.

Syntax:

```
show config|eee|statistics
show info <port-list>
```

Parameter:

config

Shows LLDP configuration

eee

Shows LLDP neighbours EEE information

statistics

Shows LLDP statistics

info

Shows LLDP neighbor device information

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(lldp)# show config
Interval      : 30
Hold          : 4
Tx Delay      : 2
Reinit Delay: 2

Port Mode      Port      System  System  System  Management CDP
Description    Name    Description Capability Address
awareness
-----
1 Disabled Enabled   Enabled Enabled Enabled Enabled
Disabled
2 Disabled Enabled   Enabled Enabled Enabled Enabled
Disabled

Switch(lldp)# show info 1
No LLDP entries found

Switch(lldp)# show statistics
LLDP global counters
Neighbor entries was last changed at 2014-01-01 00:00:00 (8222 sec. ago).
Total Neighbors Entries Added  0.
Total Neighbors Entries Deleted 0.
Total Neighbors Entries Dropped 0.
Total Neighbors Entries Aged Out 0.

LLDP local counters
Port      Rx      Tx      Rx      Rx      Rx TLV  Rx TLV  Rx TLV
Frames    Frames  Errors  Discards Errors  Unknown  Organiz. Aged
```

1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0

27 LLDP Media Commands

Media Endpoint Discovery is an enhancement of LLDP, known as LLDP-MED, that provides the following facilities:

Auto-discovery of LAN policies (such as VLAN, Layer 2 Priority and Differentiated services (Diffserv) settings) enabling plug and play networking.

Device location discovery to allow creation of location databases and, in the case of Voice over Internet Protocol (VoIP), Enhanced 911 services.

Extended and automated power management of Power over Ethernet (PoE) end points.

Inventory management, allowing network administrators to track their network devices, and determine their characteristics (manufacturer, software and hardware versions, serial or asset number).

This section allows you to configure the LLDP-MED. This function applies to VoIP devices which support LLDP-MED.

Command	Function
civic	Configure LLDP-MED civic address location
coordinate	Configure LLDP-MED coordinate location
delete	Delete the selected policy
ecs	Configure LLDP-MED Emergency Call Service
fast	Configure LLDP-MED fast start repeat count
policy	Configure LLDP-MED policy
port-policy	Configure LLDP-MED port policy
show	Show LLDP-MED information

27.1 *civic*

Using this command you can configure LLDP-MED civic address location function.

Syntax:

```
civic additional-code <LINE>
civic additional-info <LINE>
civic apartment <LINE>
civic block <LINE>
civic city <LINE>
civic ...
civic zip_code <LINE>
```

Parameter:

additional-code

Additional code

additional-info

Additional location info

apartment

Unit (Apartment, suite)

block

Neighbourhood, block

building

Building (structure)

city

City, township, shi (Japan)

comm-name

Postal community name

country-code

The two-letter ISO 3166 country code

county

County, parish, gun (Japan), district

district

City division, borough, city district, ward, chou(Japan)

floor

Floor

house-no

House number

house-no-suffix

House number suffix

landmark

Landmark or vanity address

leading-street-direction

Leading street direction

name

Name (residence and office occupant)

p.o.box

Post office box (P.O. BOX)

place-type

Place type

room-number

Room number

state

National subdivisions (state, canton, region, province, prefecture)

street

Street

street-suffix

Street suffix

trailing-street-suffix

Trailing street suffix

zip_code

Postal/zip code

<LINE>

The value for the Civic Address Location entry

Example:

```

Switch(lldpmed)# civic city taipei
Switch(lldpmed)# civic floor 1
Switch(lldpmed)# show config

Fast Start Repeat Count    : 4

Location Coordinates
-----
Latitude                   : 0.0000 North
Longitude                  : 0.0000 East
Altitude                   : 0.0000 meter(s)
Map datum                  : WGS84

Civic Address Location
-----
Country code               :
National subdivison        :
County                     :
City                       : taipei
City district              :
Block (Neighborhood)       :
Street                     :
Street Dir                 :
Trailling Street            :
Street Suffix              :
House No.                  :
House No. Suffix           :
Landmark                   :
Additional Location Info   :
Name                       :
Zip                         :
Building                   :
Unit                       :
Floor                      : 1
Room No.                   :
Placetype                  :
Postal Community Name      :
P.O. Box                   :
Addination Code            :

Emergency Call Service     :

```

27.2 coordinate

Using this command you can configure LLDP-MED coordinate location function.

Syntax:

```
coordinate altitude <coordinate-value> floor|meter
coordinate datum nad83-mlw|nad83-navd88|wgs84
coordinate latitude <coordinate-value> north|south
coordinate longitude <coordinate-value> east|west
```

Parameter:**altitude**

Altitude

<coordinate-value>

-32767 to 32767 Meters or floors with max. 4 digits00

floor

Representing altitude in a form more relevant in buildings which have different floor-to-floor dimensions

meter

Representing meters of Altitude defined by the vertical datum specified

datum

Map datum

nad83-mlw

North American Datum 1983, CRS Code 4269, Prime Meridian

Name: Greenwich; The associated vertical datum is Mean Lower Low Water (MLLW).

This datum pair is to be used when referencing locations on water/sea/ocean.

nad83-navd88

North American Datum 1983, CRS Code 4269, Prime Meridian

Name: Greenwich; The associated vertical datum is the North American Vertical Datum of 1988 (NAVD88).

This datum pair is to be used when referencing locations on land, not near tidal water (which would use Datum = NAD83/MLLW)

wgs84

(Geographical 3D) - World Geodesic System 1984, CRS Code 4327, Prime Meridian

Name: Greenwich

latitude

Latitude

<coordinate-value>

0 to 90 degrees with max. 4 digits

north

North of the equator

south

South of the equator

longitude

Longitude

<coordinate-value>

0 to 180 degrees with max. 4 digits

east

East of the prime meridian

west

West of the prime meridian

Example:

```
Switch(lldpmed)# coordinate altitude 10 floor
Switch(lldpmed)# coordinate datum nad83-mllw
Switch(lldpmed)# coordinate latitude 60 north
Switch(lldpmed)# coordinate longitude 30 east
Switch(lldpmed)# show config

Fast Start Repeat Count    : 4

Location Coordinates
-----
Latitude                   : 60.0000 North
Longitude                  : 30.0000 East
Altitude                   : 10.0000 floor
Map datum                  : NAD83/MLLW
```

27.3 delete

Using this command you can delete the selected policy.

Syntax:

```
delete <0-31>
```

Parameter:

<0-31>

Policy ID, available value is from 0 to 31

Example:

```
Switch(lldpmed)# delete 1
Switch(lldpmed)# show policy
Policy Id  Application Type      Tag      Vlan ID  L2 Priority  DSCP
-----  -
```

27.4 ecs

Using this command you can configure LLDP-MED Emergency Call Service.

Syntax:

```
ecs <number>
```

Parameter:**<number>**

The numerical digit string for the Emergency Call Service

Example:

```
Switch(lldpmed)# ecs 0921555678
Switch(lldpmed)# show config

Fast Start Repeat Count    : 4

Location Coordinates
-----
Latitude                   : 60.0000 North
Longitude                  : 30.0000 East
Altitude                   : 10.0000 floor
Map datum                  : NAD83/MLLW

Emergency Call Service     : 0921555678
```

27.5 *fast*

Using this command you can configure LLDP-MED fast start repeat count function.

Syntax:

```
fast <1-10>
```

Parameter:**<1-10>**

The number of times the fast start LLDPDU are being sent during the activation of the fast start mechanism defined by LLDP-MED

Example:

```
Switch(lldpmed)# fast 10
Switch(lldpmed)# show config

Fast Start Repeat Count    : 10

Location Coordinates
-----
Latitude                   : 60.0000 North
Longitude                  : 30.0000 East
Altitude                   : 10.0000 floor
Map datum                  : NAD83/MLLW
```

27.6 *policy*

Using this command you can configure LLDP-MED policy.

Syntax:

```
policy tagged|untagged <1-4094> <0-7> <0-63> <classification>
```

Parameter:

tagged

The device is using tagged frames

untagged

The device is using untagged frames

<1-4094>

VLAN ID, possible values from 1 to 4094.

<0-7>

Layer 2 priority to be used for the specified application type

<0-63>

DSCP value to be used to provide Diffserv node behaviour for the specified application type as defined in IETF RFC 2474

classification

guest-voice

Guest Voice to support a separate limited feature-set voice service for guest users and visitors with their own IP Telephony handsets and other similar appliances supporting interactive voice services

guest-voice-signaling

Guest Voice Signaling (conditional) for use in network topologies that require a different policy for the guest voice signaling than for the guest voice media

softphone-voice

Softphone Voice for use by softphone applications on typical data centric devices, such as PCs or laptops. This class of endpoints frequently does not support multiple VLANs, if at all, and are typically configured to use an untagged VLAN or a single tagged data specific VLAN

streaming-video

Streaming Video for use by broadcast or multicast based video content distribution and other similar applications supporting streaming video services that require specific network policy treatment. Video applications relying on TCP with buffering would not be an intended use of this application type

video-conferencing

Video Conferencing for use by dedicated Video Conferencing equipment and other similar appliances supporting real-time interactive video/audio services

video-signaling

Video Signaling (conditional) for use in network topologies that require a separate policy for the video signaling than for the video media

voice

Voice for use by dedicated IP Telephony handsets and other similar appliances supporting interactive voice services. These devices are typically deployed on a separate VLAN for ease of deployment and enhanced security by isolation from data applications

voice-signaling

Voice Signaling (conditional) for use in network topologies that require a different policy for the voice signaling than for the voice media

Example:

```
Switch(lldpmed)# policy tagged 1 0 60 guest-voice
New policy added with policy id: 1
Switch(lldpmed)# show policy
```

Policy Id	Application Type	Tag	Vlan ID	L2 Priority	DSCP
0	Guest Voice	Tagged	1	0	60

27.7 port-policy

Using this command you can configure LLDP-MED port policy function.

Syntax:

```
port-policy <port-list> <0-31> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-31>

Policy ID, available value is from 0 to 31

disable

Disables the policy to a given port

enable

Enables the policy to a given port

Example:

```
Switch(lldpmed)# port-policy 1 2 enable
Switch(lldpmed)# show port-policy
```

Port	Policies
1	2
2	none
3	none
4	none
5	none

27.8 show

Using this command you can display LLDP-MED information.

Syntax:

```
show config|policy|port-policy
show info <port-list>
```

Parameter:

config

Shows LLDP-MED configuration

policy

Shows LLDP-MED policy configuration

port-policy

Shows LLDP-MED port policy configuration

info

Shows LLDP-MED neighbor device information

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
witch(lldpmed)# show config

Fast Start Repeat Count    : 10

Location Coordinates
-----
Latitude                   : 60.0000 North
Longitude                  : 30.0000 East
Altitude                   : 10.0000 floor
Map datum                  : NAD83/MLLW

Civic Address Location
-----
Country code               :
National subdivision       :
County                     :
City                       : taipei
City district              :
Block (Neighborhood)       :
Street                     :
Street Dir                 :
Trailling Street           :
Street Suffix              :
House No.                  :
House No. Suffix           :
Landmark                   :
Additional Location Info   :
Name                       :
Zip                         :
```

```
Building          :
Unit              :
Floor             : 1
Room No.          :
Placetype         :
Postal Community Name :
P.O. Box          :
Addination Code   :

Emergency Call Service : 0921555678

Switch(lldpmed)# show info 1
No LLDP-MED entries found

Switch(lldpmed)# show policy

```

Policy Id	Application Type	Tag	Vlan ID	L2 Priority	DSCP
0	Guest Voice	Tagged	1	0	60

```
Switch(lldpmed)# show port-policy
Port    Policies
-----
1       2
2       none
3       none
```

28 Loop protection Commands

The loop detection is used to detect the presence of traffic. When switch receives packet's (looping detection frame) MAC address the same as oneself from port, show Loop Protection happens. The port will be locked when it received the looping detection frames.

Command	Function
interval	Configure loop protection transmit interval
mode	Configure loop protection mode
port-action	Configure loop protection port action
port-mode	Configure loop protection port mode
port-transmit	Configure loop protection port transmit mode
show	Display loop protection information
shutdown	Configure loop protection shutdown time

28.1 *interval*

Using this command you can configure loop protection transmit interval.

Syntax:

```
interval <1-10>
```

Parameter:

<1-10>

Transmit time interval

Example:

```
Switch(loop-protect)# interval 3
Switch(loop-protect)# show config
Loop Protection    : Disabled
Transmission Time  : 3
Shutdown Time     : 180
```

28.2 *mode*

Using this command you can configure loop protection mode globally.

Syntax:

```
mode disable|enable
```

Parameter:**disable**

Disables loop protection mode globally.

enable

Enables loop protection mode globally.

Example:

```
Switch(loop-protect)# mode enable
Switch(loop-protect)# show config
Loop Protection      : Enabled
Transmission Time   : 3
Shutdown Time       : 180
```

Port	Mode	Action	Transmit
1	Enabled	Shutdown	Enabled
2	Enabled	Shutdown	Enabled
3	Enabled	Shutdown	Enabled
4	Enabled	Shutdown	Enabled

28.3 port-action

Using this command you can configure loop protection port action.

Syntax:

```
port-action <port-list> both|log|shutdown
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

both

Shutdown the port and log event.

log

Log the event only.

shutdown

Shutdown the port.

Example:

```
Switch(loop-protect)# port-action 1 both
Switch(loop-protect)# show config
Loop Protection      : Disabled
Transmission Time   : 3
Shutdown Time       : 180
```

Port	Mode	Action	Transmit
------	------	--------	----------

```

-----
1      Enabled  Shutdown and Log  Enabled
2      Enabled  Shutdown           Enabled

Switch(loop-protect)# port-action 1 log
Switch(loop-protect)# show config
Loop Protection   : Disabled
Transmission Time : 3
Shutdown Time    : 180

Port  Mode      Action           Transmit
-----
1      Enabled  Log Only           Enabled
2      Enabled  Shutdown           Enabled

Switch(loop-protect)# port-action 1 shutdown
Switch(loop-protect)# show config
Loop Protection   : Disabled
Transmission Time : 3
Shutdown Time    : 180

Port  Mode      Action           Transmit
-----
1      Enabled  Shutdown           Enabled
2      Enabled  Shutdown           Enabled

```

28.4 port-mode

Using this command you can configure loop protection port mode for each port.

Syntax:

```
port-mode <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables loop protection mode for the selected port(s).

enable

Enables loop protection mode for the selected port(s).

Example:

```

Switch(loop-protect)# port-mode 1 disable
Switch(loop-protect)# show config
Loop Protection   : Disabled
Transmission Time : 3
Shutdown Time    : 180

```

```

Port  Mode      Action           Transmit
-----

```

1	Disabled	Shutdown	Enabled
2	Enabled	Shutdown	Enabled
3	Enabled	Shutdown	Enabled

28.5 *port-transmit*

Using this command you can configure loop protection port transmit mode for each port.

Syntax:

```
port-transmit <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Passively looking for looped PDU's on the selected port(s).

enable

Actively generating loop protection PDU's on the selected port(s).

Example:

```
Switch(loop-protect)# port-transmit 1 disable
Switch(loop-protect)# show config
Loop Protection    : Disabled
Transmission Time  : 3
Shutdown Time     : 180
```

Port	Mode	Action	Transmit
1	Disabled	Shutdown	Disabled
2	Enabled	Shutdown	Enabled

28.6 *show*

The command display loop protection information.

Syntax:

```
show config|status
```

Parameter:

config

Shows loop protection configuration.

status

Shows loop protection status.

Example:

```
Switch(loop-protect)# show config
Loop Protection   : Disabled
Transmission Time : 5
Shutdown Time    : 180

Port  Mode      Action          Transmit
----  -
1     Enabled    Shutdown         Enabled
2     Enabled    Shutdown         Enabled
3     Enabled    Shutdown         Enabled
4     Enabled    Shutdown         Enabled

Switch(loop-protect)# show status
Port Action          Transmit  Loops   Status  Loop  Time of Last
Loop
-----
1     Shutdown         Enabled   0       Down    -     -
2     Shutdown         Enabled   0       Down    -     -
3     Shutdown         Enabled   0       Down    -     -
4     Shutdown         Enabled   0       Down    -     -
```

28.7 shutdown

Using this command you can configure loop protection shutdown time.

Syntax:

```
shutdown <0-604800>
```

Parameter:

<0-604800>

Shutdown time interval. A value of zero disables re-enabling the port.

Example:

```
Switch(loop-protect)# shutdown 200
Switch(loop-protect)# show config
Loop Protection   : Disabled
Transmission Time : 3
Shutdown Time    : 200
```

29 Port Mirroring Commands

You can mirror traffic from any source port to a target port for real-time analysis. You can then attach a logic analyzer or RMON probe to the target port and study the traffic crossing the source port in a completely unobtrusive manner.

Mirror Configuration is to monitor the traffic of the network. For example, we assume that Port A and Port B are Monitoring Port and Monitored Port respectively, thus, the traffic received by Port B will be copied to Port A for monitoring.

Command	Function
analyzer-port	Configure analyzer port
port-mode	Configure port mode
show	Show port mirroring information

29.1 *analyzer-port*

Using this command you can configure the analyzer port on the switch.

Syntax:

```
analyzer-port <port>|disable
```

Parameter:

disable

Disables port mirroring

<port>

Defines this port as analyzer port and enables port mirroring, possible values depending on distinct hardware model.

Example:

```
Switch(mirror)# analyzer-port 1
Switch(mirror)# show

Analyzer Port: 1

Port  Mode
----  -
1     Disabled
2     Disabled
```

29.2 *port-mode*

Using this command you can configure port mirroring mode for each port.

Syntax:

```
port-mode <port-list> disable|enable|rx-only|tx-only
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

The parameter means you to disable DHCP relay mode.

enable

The parameter means you to enable DHCP snooping mode.

rx-only

Enables Rx mirroring on the selected port(s).

tx-only

Enables Tx mirroring on the selected port(s).

Example:

```
Switch(mirror)# port-mode 2 enable
Switch(mirror)# port-mode 3 rx-only
Switch(mirror)# port-mode 4 tx-only
Switch(mirror)# port-mode 1 disable
Switch(mirror)# show
```

Analyzer Port: 1

Port	Mode
1	Disabled
2	Enabled
3	Rx-only
4	Tx-only

29.3 show

Using this command you can show port mirroring information.

Syntax:

```
show
```

Example:

```
Switch(mirror)# show
```

Analyzer Port: Disabled

Port	Mode
1	Disabled
2	Disabled

3	Disabled
4	Disabled

30 MLD Commands

Curiously enough, a network node that acts as a source of IPv6 multicast traffic is only an indirect participant in MLD snooping – it just provides multicast traffic, and MLD doesn't interact with it. (Note, however, that in an application like desktop conferencing a network node may act as both a source and an MLD host; but MLD interacts with that node only in its role as an MLD host.)

A source node creates multicast traffic by sending packets to a multicast address. In IPv6, addresses with the first eight bits set (that is, "FF" as the first two characters of the address) are multicast addresses, and any node that listens to such an address will receive the traffic sent to that address. Application software running on the source and destination systems cooperates to determine what multicast address to use. (Note that this is a function of the application software, not of MLD.)

When MLD snooping is enabled on a VLAN, the switch acts to minimize unnecessary multicast traffic. If the switch receives multicast traffic destined for a given multicast address, it forwards that traffic only to ports on the VLAN that have MLD hosts for that address. It drops that traffic for ports on the VLAN that have no MLD hosts.

Command	Function
compatibility	Set the Versions of MLD Operating on Hosts and Routers
delete	Delete commands
fast-leave	Set per-port Fast Leave
filtering	The IP Multicast Group that will be filtered
flooding	Set MLD Flooding Mode
lmqi	Set the per-VLAN Last Member Query Interval
proxy	Set MLD Proxy Mode
qi	Set the per-VLAN Query Interval
qri	Set the per-VLAN Query Response Interval
querier	Enable/Disable the per-VLAN MLD Querier
router	Set Router Port
rv	Set the per-VLAN Robustness Variable
show	Show MLD Information
snooping	Set MLD Snooping Mode
ssm-range	Set MLD SSM Range
state	Enable/Disable the per-VLAN MLD Snooping
throttling	Set per-port Throttling
uri	Set the per-VLAN Unsolicited Report Interval

30.1 *compatibility*

Using this command you can set the Versions of MLD Operating on Hosts and Routers.

Syntax:

```
compatibility <vlan-list> Forced-MLDv1|Forced-MLDv2|MLD-Auto
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

Forced-MLDv1

Set MLDv1 of MLD operating on hosts and routers

Forced-MLDv2

Set MLDv2 of MLD operating on hosts and routers

MLD-Auto

Set auto mode of MLD operating on hosts and routers

Example:

```
Switch(mld)# compatibility 1 forced-MLDv1
```

30.2 delete

Using this command you can delete the MLD filtering group.

Syntax:

```
delete <port-list> <ipv6-address>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<ipv6-address>

IP Multicast Group to delete.

Example:

```
Switch(mld)# delete 1 fe80::202:b3ff:fe1e:8329
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Enabled
MLd Proxy : Disabled
```

30.3 fast-leave

Using this command you can configure Fast Leave for each port.

Syntax:

```
fast-leave <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables fast leave on the selected port(s).

enable

Enables fast leave on the selected port(s)

Example:

```
Switch(mld)# fast-leave 1 enable
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Enabled
MLd Proxy : Disabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
----  -
1     Disabled      No              Enabled     Unlimited
2     Disabled      No              Disabled    Unlimited
3     Disabled      No              Disabled    Unlimited
4     Disabled      No              Disabled    Unlimited
```

30.4 filtering

Using this command you can set the IP Multicast Group that will be filtered.

Syntax:

```
filtering <port-list> <ipv6-address>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<ipv6-address>

IPv6 address is in 128-bit records represented as eight fields of up to four hexadecimal digits with a colon separate each field (":"). For example, **fe80::215:c5ff:fe03:4dc7**. The symbol "::" is a special syntax that can be used as a shorthand way of representing multiple 16-bit groups of contiguous zeros; but it can only appear once. It also used a following legally IPv4 address. For example, **::192.1.2.34**.

Example:

```
Switch(mld)# filtering 1 fe80::215:c5ff:fe03:4dc7
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Enabled
MLd Proxy : Disabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
-----
1      Disabled    No              Enabled      Unlimited
2      Disabled    No              Disabled     Unlimited
3      Disabled    No              Disabled     Unlimited
4      Disabled    No              Disabled     Unlimited

VID   State      Querier  Compatibility  RV   QI    QRI    LLQI    URI
-----
1      Disabled  Enabled

Port  Filtering Groups
-----
1      6665:3830:3a3a:3231:353a:6335:6666:3a66
2      No Filtering Group
3      No Filtering Group
```

30.5 flooding

Using this command you can set MLD Flooding Mode.

Syntax:

```
flooding disable|enable
```

Parameter:**disable**

Disables unregistered IPMCv6 traffic flooding.

enable

Enables unregistered IPMCv6 traffic flooding.

Example:

```
Switch(mld)# flooding disable
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Disabled
MLd Proxy : Disabled
```

30.6 *lmqi*

Using this command you can set the per-VLAN Last Member Query Interval.

Syntax:

```
lmqi <vlan-list> <0-31744>
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<0-31744>

Range:0~31744 tenths of sec, default:100 tenths of sec.

Example:

```
Switch(mld)# lmqi 1 31744
```

30.7 *proxy*

Using this command you can set MLD Proxy Mode.

Syntax:

```
proxy disable|enable
```

Parameter:**disable**

Disables MLD proxy.

enable

Enables MLD proxy.

Example:

```
Switch(mld)# proxy enable
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Disabled
MLd Proxy : Enabled
```

30.8 *qi*

Using this command you can set the per-VLAN Query Interval.

Syntax:

```
qi <vlan-list> <1-31744>
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<1-31744>

Range:1~31744 sec, default:125 sec

Example:

```
Switch(mld)# state 1 enable
Switch(mld)# qi 1 888
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Enabled
MLd Proxy : Disabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
----  -
1      Disabled No              Disabled    Unlimited
2      Disabled No              Disabled    Unlimited

VID   State   Querier  Compatibility  RV   QI    QRI    LLQI   URI
----  -
1      Enabled En
```

30.9 *qri*

Using this command you can set the per-VLAN Query Response Interval.

Syntax:

```
qri <vlan-list> <0-31744>
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<0-31744>

Range:0~31744 tenths of sec, default:100 tenths of sec.

Example:

```
Switch(mld)# state 1 enable
Switch(mld)# qri 1 555
Switch(mld)# show config
MLD Snooping : Disabled
```

```

MLd Flooding Control : Enabled
MLd Proxy : Disabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
-----
1      Disabled    No              Disabled    Unlimited
2      Disabled    No              Disabled    Unlimited

VID   State      Querier      Compatibility  RV   QI    QRI    LLQI    URI
-----
1      Enabled    Enabled      IGMP-Auto      2    888   555    10     1

```

30.10 querier

Using this command you can Enable/Disable the per-VLAN MLD Querier.

Syntax:

```
querier <vlan-list> disable|enable
```

Parameter:

<vlan-list>

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

disable

Disables the per-VLAN MLD querier.

enable

Enables the per-VLAN MLD querier.

Example:

```

Switch(mld)# querier 1 enable
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Enabled
MLd Proxy : Disabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
-----
1      Disabled    No              Disabled    Unlimited
2      Disabled    No              Disabled    Unlimited

VID   State      Querier      Compatibility  RV   QI    QRI    LLQI    URI
-----
1      Enabled    Enabled      IGMP-Auto      99   888   555    10     1

```

30.11 *router*

Using this command you can set Router Port.

Syntax:

```
router <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables router port

enable

Enables router port.

Example:

```
Switch(mld)# router 1 enable
Switch(mld)# show config
MLD Snooping : Enabled
MLd Flooding Control : Disabled
MLd Proxy : Enabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
-----
1      Enabled      No             Disabled    Unlimited
2      Disabled     No             Disabled    Unlimited
3      Disabled     No             Disabled    Unlimited
4      Disabled     No             Disabled    Unlimited
```

30.12 *rv*

Using this command you can set the per-VLAN Robustness Variable.

Syntax:

```
rv <vlan-list> <2-255>
```

Parameter:

<vlan-list>

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<2-255>

Range:2~255, default:2.

Example:

```
Switch(mld)# rv 1 99
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Enabled
MLd Proxy : Disabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
-----
1      Disabled      No              Disabled     Unlimited
2      Disabled      No              Disabled     Unlimited

VID   State      Querier      Compatibility  RV   QI    QRI    LLQI    URI
-----
1      Enabled     Enabled      IGMP-Auto      99   888   555    10     1
```

30.13 *show*

Using this command you can show MLD Information.

Syntax:

```
show config
show groups|ssm|status|version [<1-4094>]
```

Parameter:**config**

Shows MLD Configuration

groups

Entries in the MLD Group table

ssm

Entries in the MLDv2 information table

status

Shows MLD status

version

Shows MLD working querier/host version currently

<1-4094>

VLAN ID, possible values from 1 to 4094.

Example:

```
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Enabled
MLd Proxy : Disabled

MLD SSM Range: ff3e::/96
```

Port	Router	Dynamic Router	Fast Leave	Group Throttling Number
1	Disabled	No	Disabled	Unlimited
2	Disabled	No	Disabled	Unlimited
3	Disabled	No	Disabled	Unlimited
4	Disabled	No	Disabled	Unlimited

VID	State	Querier	Compatibility	RV	QI	QRI	LLQI	URI
1	Disabled	Enabled						

Port	Filtering Groups
1	No Filtering Group
2	No Filtering Group
3	No Filtering Group
4	No Filtering Group

30.14 *snooping*

Using this command you can set MLD Snooping Mode.

Syntax:

```
snooping disable|enable
```

Parameter:

disable

Disables the global MLD snooping

enable

Enables the global MLD snooping

Example:

```
Switch(mld)# snooping enable
Switch(mld)# show config
MLD Snooping : Enabled
MLd Flooding Control : Disabled
MLd Proxy : Enabled
```

30.15 *ssm-range*

Using this command you can set MLD SSM Range.

Syntax:

```
ssm-range <ipv6-address> <8-128>
```

Parameter:**<ipv6-address>**

Set MLD SSM range address.

<8-128>

Set MLD SSM range value.

Example:

```
ssm-range ::ffff:192.168.1.6 10
```

30.16 *state*

Using this command you can Enable/Disable the per-VLAN MLD Snooping.

Syntax:

```
state <vlan-list> disable|enable
```

Parameter:**<vlan-list>**

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

disable

Disables the per-VLAN MLD snooping

enable

Enables the per-VLAN MLD snooping

Example:

```
Switch(mld)# state 1 enable
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Enabled
MLd Proxy : Disabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
----  -
1     Disabled      No              Disabled     Unlimited
2     Disabled      No              Disabled     Unlimited

VID   State   Querier  Compatibility  RV   QI    QRI    LLQI   URI
----  -
1     Enabled  Enabled  IGMP-Auto      99   888   555    10     1
```

30.17 *throttling*

Using this command you can set per-port Throttling.

Syntax:

```
throttling <port-list> <0-10>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-10>

Set port group limit number, range:0~10, 0:unlimited

Example:

```
witch(mld)# throttling 1 10
Switch(mld)# show config
MLD Snooping : Enabled
MLd Flooding Control : Disabled
MLd Proxy : Enabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
----  -
1      Disabled No             Disabled    10
2      Disabled No             Disabled    Unlimited
3      Disabled No             Disabled    Unlimited
4      Disabled No             Disabled    Unlimited
```

30.18 *uri*

Using this command you can set the per-VLAN Unsolicited Report Interval.

Syntax:

```
uri <vlan-list> <0-31744>
```

Parameter:

<vlan-list>

VLAN list, possible values from 1 to 4094. Individual VLANs are separated by comma, VLAN ranges are joined by hyphen (1,3-5).

<0-31744>

Range:0~31744 sec, default:1 sec

Example:

```
Switch(mld)# uri 1 777
Switch(mld)# show config
MLD Snooping : Disabled
MLd Flooding Control : Enabled
MLd Proxy : Disabled

MLD SSM Range: ff3e::/96
Port  Router      Dynamic Router  Fast Leave  Group Throttling Number
-----
1     Disabled    No              Disabled     Unlimited
2     Disabled    No              Disabled     Unlimited

VID   State      Querier  Compatibility  RV   QI    QRI    LLQI  URI
-----
1     Enabled    Enabled  IGMP-Auto      99   888   555    10    777
```

31 MVR Commands

The MVR feature enables multicast traffic forwarding on the Multicast VLAN. In a multicast television application, a PC or a television with a set-top box can receive the multicast stream. Multiple set-top boxes or PCs can be connected to one subscriber port, which is a switch port configured as an MVR receiver port. When a subscriber selects a channel, the set-top box or PC sends an IGMP join message to Switch A to join the appropriate multicast. Uplink ports that send and receive multicast data to and from the multicast VLAN are called MVR source ports.

Command	Function
immediate-leave	Configure MVR port state about immediate leave
mode	Configure MVR mode
port-mode	Configure MVR port mode
port-type	Configure MVR port type
show	Show command

31.1 *allow*

The IP Multicast Group that will be allowed.

Syntax:

```
allow <port-list> <ipmc-address> <ipmc-address>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<ipmc-address>

Add new entry to the group allow table, group allow start address

<ipmc-address>

Add new entry to the group allow table, group allow end address

Example:

```
Switch(mvr)# allow 5 224.0.0.2 224.0.0.5
Switch(mvr)#
```

31.2 *delete*

Delete MVR allow group.

Syntax:

```
delete <port-list> <ipmc-address> <ipmc-address>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<ipmc-address>

Delete MVR allow group, group allow start address

<ipmc-address>

Delete MVR allow group, group allow end address

Example:

```
Switch(mvr)# delete 5 224.0.0.2 224.0.0.5
Switch(mvr)#
```

31.3 *immediate-leave*

Using this command you can configure MVR port state about immediate leave.

Syntax:

```
immediate-leave <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables immediate leave on the specific port

enable

Enables immediate leave on the specific port

Example:

```
Switch(mvr)# immediate-leave 1 enable
Switch(mvr)# show config
MVR Mode      : Disabled
Multicast VLAN ID : 100
```

Port	Port Mode	Port Type	Immediate Leave
1	Disabled	Receive	Enabled
2	Disabled	Receive	Disabled
3	Disabled	Receive	Disabled
4	Disabled	Receive	Disabled

31.4 *mode*

Using this command you can configure MVR mode.

Syntax:

```
mode disable|enable <1-4094>
```

Parameter:**disable**

Disables MVR

enable

Enables multicast traffic forwarding on the Multicast VLAN

<1-4094>

Multicast VLAN ID, available is from 1 to 4094

Example:

```
Switch(mvr)# mode enable 1
Switch(mvr)# show config
MVR Mode      : Enabled
Muticast VLAN ID : 1
```

31.5 *port-mode*

Using this command you can configure MVR port mode.

Syntax:

```
port-mode <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables MVR on the specific port

enable

Enables MVR on the specific port

Example:

```
Switch(mvr)# port-mode 1 enable
Switch(mvr)# show config
MVR Mode      : Disabled
Muticast VLAN ID : 1
```

Port	Port Mode	Port Type	Immediate Leave
-----	-----	-----	-----
1	Enabled	Receive	Enabled
2	Disabled	Receive	Disabled
3	Disabled	Receive	Disabled
4	Disabled	Receive	Disabled

31.6 *port-type*

Using this command you can configure MVR port type.

Syntax:

```
port-type <port-list> receiver|source
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

receiver

Define the port as receiver port

source

Define the port as source port

Example:

```
witch(mvr)# port-type 2 source
Switch(mvr)# show config
MVR Mode      : Disabled
Multicast VLAN ID : 1

Port  Port Mode  Port Type  Immediate Leave
-----
1     Enabled    Receive    Enabled
2     Disabled    Source     Disabled
3     Disabled    Receive    Disabled
4     Disabled    Receive    Disabled
```

31.7 *show*

Using this command you can show command.

Syntax:

```
show allow|config|group|statistics
```

Parameter:**allow**

Shows MVR allow group information

config

Shows MVR configuration

group

Shows MVR group information

statistics

Shows MVR statistics information

Example:

```
Switch(mvr)# show config
MVR Mode      : Disabled
Multicast VLAN ID : 100

Port  Port Mode  Port Type  Immediate Leave
-----
1     Disabled   Receive    Disabled
2     Disabled   Receive    Disabled
3     Disabled   Receive    Disabled
4     Disabled   Receive    Disabled

Switch(mvr)# show group

Switch(mvr)# show statistics
```

32 NAS Commands

The section describes to configure the Network Access Server parameters of the switch. The NAS server can be employed to connect users to a variety of resources including Internet access, conference calls, printing documents on shared printers, or by simply logging on to the Internet.

Command	Function
<code>agetime</code>	Configure the time in seconds between check for activity on successfully authenticated MAC addresses
<code>clear</code>	Clear NAS statistics
<code>eapol-timeout</code>	Configure the time between EAPOL retransmissions
<code>guest-vlan</code>	Configure the Guest VLAN mode
<code>hold-time</code>	Configure the time in seconds before a MAC-address that failed authentication gets a new authentication chance
<code>mode</code>	Configure the NAS mode
<code>port-guest-vlan</code>	Configure the Guest VLAN mode of switch ports
<code>port-radius-qos</code>	Configure the RADIUS-assigned QoS mode of switch ports
<code>port-radius-vlan</code>	Configure the RADIUS-assigned VLAN mode of switch ports
<code>port-state</code>	Configure the NAS port state
<code>radius-qos</code>	Configure the RADIUS-assigned QoS mode
<code>radius-vlan</code>	Configure the RADIUS-assigned VLAN mode
<code>reauth-period</code>	Configure the period between reauthentications
<code>reauthentication</code>	Configure the NAS reauthentication mode
<code>restart</code>	Restart NAS authentication process
<code>show</code>	Show NAS information

32.1 *agetime*

Using this command you can configure the time in seconds between check for activity on successfully authenticated MAC addresses.

Syntax:

```
agetime <10-1000000>
```

Parameter:

<10-1000000>

Time in seconds between checks for activity on a MAC address that succeeded authentication

Example:

```
Switch(nas)# agetime 9999
Switch(nas)# show config
```

```

Mode                               : Disabled
Reauthentication                   : Disabled
Reauthentication Period            : 3600
EAPOL Timeout                      : 30
Age Period                        : 9999
Hold Time                         : 10
RADIUS QoS                        : Disabled
RADIUS VLAN                       : Disabled
Guest VLAN                       : Disabled
Guest VLAN ID                     : 1
Maximum Reauthentication Count     : 2
Allow Guest VLAN if EAPOL Frame Seen : Disabled

```

32.2 *clear*

Using this command you can clear NAS statistics.

Syntax:

```
clear <port-list>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(nas)# clear 1
```

32.3 *eapol-timeout*

Using this command you can configure the time between EAPOL retransmissions.

Syntax:

```
eapol-timeout <1-65535>
```

Parameter:

<1-65535>

Time in seconds between EAPOL retransmissions

Example:

```

Switch(nas)# eapol-timeout 8888
Switch(nas)# show config
Mode                               : Disabled
Reauthentication                   : Disabled

```

```

Reauthentication Period      : 3600
EAPOL Timeout               : 8888
Age Period                  : 9999
Hold Time                   : 10
RADIUS QoS                  : Disabled
RADIUS VLAN                 : Disabled
Guest VLAN                  : Disabled
Guest VLAN ID               : 1
Maximum Reauthentication Count : 2
Allow Guest VLAN if EAPOL Frame Seen : Disabled

```

32.4 *guest-vlan*

Using this command you can configure the Guest VLAN mode.

Syntax:

```

guest-vlan disable
guest-vlan enable <1-4094> <1-255> allow_if_eapol_seen (disable|enable)

```

Parameter:

disable

Disables Guest VLAN

enable

Enables Guest VLAN

<1-4094>

Guest VLAN ID used when entering the Guest VLAN

<1-255>

The number of times a Request Identity EAPOL frame is sent without response before considering entering the Guest VLAN

allow_if_eapol_seen

The switch remembers if an EAPOL frame has been received on the port for the life-time of the port. Once the switch considers whether to enter the Guest VLAN, it will first check if this option is enabled or disabled

disable

The Guest VLAN can only be entered if no EAPOL frames have been received on a port for the lifetime of the port

enable

The Guest VLAN can be entered even if an EAPOL frame has been received during the lifetime of the port

Example:

```

Switch(nas)# guest-vlan enable 90 150 allow_if_eapol_seen enable
Switch(nas)# show config
Mode                               : Disabled
Reauthentication                   : Disabled
Reauthentication Period            : 3600
EAPOL Timeout                     : 8888
Age Period                         : 9999
Hold Time                         : 10

```

```

RADIUS QoS           : Disabled
RADIUS VLAN          : Disabled
Guest VLAN           : Enabled
Guest VLAN ID        : 90
Maximum Reauthentication Count : 150
Allow Guest VLAN if EAPOL Frame Seen : Enabled

```

32.5 *hold-time*

Using this command you can configure the time in seconds before a MAC-address that failed authentication gets a new authentication chance.

Syntax:

```
hold-time <10-1000000>
```

Parameter:

<10-1000000>

Hold time before MAC addresses that failed authentication expire

Example:

```

Switch(nas)# hold-time 7777
Switch(nas)# show config
Mode           : Disabled
Reauthentication : Disabled
Reauthentication Period : 3600
EAPOL Timeout   : 8888
Age Period      : 9999
Hold Time       : 7777
RADIUS QoS      : Disabled
RADIUS VLAN     : Disabled
Guest VLAN      : Enabled
Guest VLAN ID   : 90
Maximum Reauthentication Count : 150
Allow Guest VLAN if EAPOL Frame Seen : Enabled

```

32.6 *mode*

Using this command you can configure the NAS mode.

Syntax:

```
mode disable|enable
```

Parameter:

disable

Globally disable NAS operation mode

enable

Globally enable NAS operation mode

Example:

```
Switch(nas)# mode enable
Switch(nas)# show config
Mode                                     : Enabled
Reauthentication                       : Disabled
Reauthentication Period                 : 3600
EAPOL Timeout                          : 8888
Age Period                             : 9999
Hold Time                              : 7777
RADIUS QoS                             : Disabled
RADIUS VLAN                            : Disabled
Guest VLAN                             : Enabled
Guest VLAN ID                          : 90
Maximum Reauthentication Count          : 150
Allow Guest VLAN if EAPOL Frame Seen    : Enabled
```

32.7 port-guest-vlan

Using this command you can configure the Guest VLAN mode of switch ports.

Syntax:

```
port-guest-vlan <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables Guest VLAN

enable

Enables Guest VLAN

Example:

```
Switch(nas)# port-guest-vlan 1 enable
Switch(nas)# show port-config
```

Port	Admin	State	RADIUS-Assigned QoS	RADIUS-Assigned VLAN	Guest VLAN
1	Force	Authorized	Disabled	Disabled	Enabled
2	Force	Authorized	Disabled	Disabled	Disabled
3	Force	Authorized	Disabled	Disabled	Disabled
4	Force	Authorized	Disabled	Disabled	Disabled

32.8 *port-radius-qos*

Using this command you can configure the RADIUS-assigned QoS mode of switch ports.

Syntax:

```
port-radius-qos <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables RADIUS-assigned QoS

enable

Enables RADIUS-assigned QoS

Example:

```
Switch(nas)# port-radius-qos 2 enable
Switch(nas)# show port-config
Port  Admin State      RADIUS-Assigned QoS  RADIUS-Assigned VLAN  Guest
VLAN
-----
1      Force Authorized    Disabled              Disabled                Enabled
2      Force Authorized    Enabled               Disabled                Disabled
3      Force Authorized    Disabled              Disabled                Disabled
4      Force Authorized    Disabled              Disabled                Disabled
```

32.9 *port-radius-vlan*

Using this command you can configure the RADIUS-assigned VLAN mode of switch ports.

Syntax:

```
port-radius-vlan <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables RADIUS-assigned VLAN

enable

Enables RADIUS-assigned VLAN

Example:

```
Switch(nas)# port-radius-vlan 3 enable
Switch(nas)# show port-config
```

Port VLAN	Admin State		RADIUS-Assigned QoS	RADIUS-Assigned VLAN	Guest
1	Force Authorized	Disabled	Disabled	Enabled	Enabled
2	Force Authorized	Enabled	Disabled	Disabled	Disabled
3	Force Authorized	Disabled	Enabled	Disabled	Disabled
4	Force Authorized	Disabled	Disabled	Disabled	Disabled

32.10 *port-state*

Using this command you can configure the NAS port state.

Syntax:

```
port-state <port-list> force-auth|force-unauth|mac-based|multi|port-based|single
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

force-auth

Port access is allowed

force-unauth

Port access is not allowed

mac-based

Switch authenticates on behalf of the client

multi

Multiple Host NAS Authentication

port-based

Port-based NAS Authentication

single

Single Host NAS Authentication

Example:

```
Switch(nas)# port-state 4 force-unauth
Switch(nas)# port-state 5 mac-based
Switch(nas)# port-state 6 multi
Switch(nas)# port-state 7 port-based
Switch(nas)# port-state 8 single
Switch(nas)# show port-config
```

Port VLAN	Admin State		RADIUS-Assigned QoS	RADIUS-Assigned VLAN	Guest
1	Force Authorized	Disabled	Disabled	Enabled	Enabled
2	Force Authorized	Enabled	Disabled	Disabled	Disabled
3	Force Authorized	Disabled	Enabled	Disabled	Disabled
4	Force Authorized	Disabled	Disabled	Disabled	Disabled

1	Force Authorized	Disabled	Disabled	Disabled	Disabled
2	Force Authorized	Disabled	Disabled	Disabled	Disabled
3	Force Authorized	Disabled	Disabled	Disabled	Disabled
4	Force Unauthorized	Disabled	Disabled	Disabled	Disabled
5	MAC-Based Auth	Disabled	Disabled	Disabled	Disabled
6	Multi 802.1X	Disabled	Disabled	Disabled	Disabled
7	Port-based 802.1X	Disabled	Disabled	Disabled	Disabled
8	Single 802.1X	Disabled	Disabled	Disabled	Disabled

32.11 *radius-qos*

Using this command you can configure the RADIUS-assigned QoS mode.

Syntax:

```
radius-qos disable|enable
```

Parameter:

disable

Disables RADIUS-assigned QoS

enable

Enables RADIUS-assigned QoS

Example:

```
Switch(nas)# radius-qos enable
Switch(nas)# show config
Mode                               : Enabled
Reauthentication                   : Disabled
Reauthentication Period            : 3600
EAPOL Timeout                      : 8888
Age Period                        : 9999
Hold Time                         : 7777
RADIUS QoS                        : Enabled
RADIUS VLAN                       : Disabled
Guest VLAN                       : Enabled
Guest VLAN ID                     : 90
Maximum Reauthentication Count     : 150
Allow Guest VLAN if EAPOL Frame Seen : Enabled
```

32.12 *radius-vlan*

Using this command you can configure the RADIUS-assigned VLAN mode.

Syntax:

```
radius-vlan disable|enable
```

Parameter:**disable**

Disables RADIUS-assigned VLAN

enable

Enables RADIUS-assigned VLAN

Example:

```
Switch(nas)# radius-vlan enable
Switch(nas)# show config
Mode                                     : Enabled
Reauthentication                       : Disabled
Reauthentication Period                 : 3600
EAPOL Timeout                          : 8888
Age Period                             : 9999
Hold Time                              : 7777
RADIUS QoS                             : Enabled
RADIUS VLAN                            : Enabled
Guest VLAN                             : Enabled
Guest VLAN ID                          : 90
Maximum Reauthentication Count          : 150
Allow Guest VLAN if EAPOL Frame Seen   : Enabled
```

32.13 *reauth-period*

Using this command you can configure the period between reauthentications.

Syntax:

```
reauth-period <1-3600>
```

Parameter:**<1-3600>**

Period between reauthentications

Example:

```
Switch(nas)# reauth-period 666
Switch(nas)# show config
Mode                                     : Enabled
Reauthentication                       : Disabled
Reauthentication Period                 : 666
EAPOL Timeout                          : 8888
Age Period                             : 9999
Hold Time                              : 7777
RADIUS QoS                             : Enabled
RADIUS VLAN                            : Enabled
Guest VLAN                             : Enabled
Guest VLAN ID                          : 90
Maximum Reauthentication Count          : 150
Allow Guest VLAN if EAPOL Frame Seen   : Enabled
```

32.14 *reauthentication*

Using this command you can configure the NAS reauthentication mode.

Syntax:

```
reauthentication disable|enable
```

Parameter:**disable**

Disables NAS reauthentication

enable

Enables NAS reauthentication

Example:

```
Switch(nas)# reauthentication enable
Switch(nas)# show config
Mode                                     : Enabled
Reauthentication                       : Enabled
Reauthentication Period                : 666
EAPOL Timeout                          : 8888
Age Period                             : 9999
Hold Time                              : 7777
RADIUS QoS                             : Enabled
RADIUS VLAN                            : Enabled
Guest VLAN                             : Enabled
Guest VLAN ID                          : 90
Maximum Reauthentication Count         : 150
Allow Guest VLAN if EAPOL Frame Seen : Enabled
```

32.15 *restart*

Using this command you can restart NAS authentication process.

Syntax:

```
restart <port-list> reauthenticate|reinitialize
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

reauthenticate

Schedules a reauthentication whenever the quiet-period of the port runs out (EAPOL-based authentication). For MAC-based authentication, reauthentication will be attempted immediately

reinitialize

Forces a reinitialization of the clients on the port and thereby a reauthentication immediately

Example:

```
Switch(nas)# restart 1 reauthenticate
```

32.16 show

Show NAS information.

Syntax:

```
show config|port-config|status
show statistics <port-list>
```

Parameter:**config**

Shows NAS configuration

port-config

Shows NAS port configuration

statistics

Shows NAS statistics

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

status

Shows NAS status

Example:

```
Switch(nas)# show config
Mode                               : Disabled
Reauthentication                   : Disabled
Reauthentication Period            : 3600
EAPOL Timeout                      : 30
Age Period                        : 300
Hold Time                         : 10
RADIUS QoS                        : Disabled
RADIUS VLAN                       : Disabled
Guest VLAN                        : Disabled
Guest VLAN ID                     : 1
Maximum Reauthentication Count     : 2
Allow Guest VLAN if EAPOL Frame Seen : Disabled

Switch(nas)# show port-config
Port  Admin State      RADIUS-Assigned QoS  RADIUS-Assigned VLAN  Guest
VLAN
-----
```

```

-----
1   Force Authorized   Disabled           Disabled           Disabled
2   Force Authorized   Disabled           Disabled           Disabled
3   Force Authorized   Disabled           Disabled           Disabled
4   Force Authorized   Disabled           Disabled           Disabled

Switch(nas)# show statistics 1

Port 1 EAPOL Statistics:
Rx Total                      0   Tx Total
0
Rx Response/Id                0   Tx Request/Id
0
Rx Response                    0   Tx Request
0
Rx Start                       0
Rx Logoff                      0
Rx Invalid Type                0
Rx Invalid Length              0

Port 1 Backend Server Statistics:
Rx Access Challenges           0   Tx Responses
0
Rx Other Requests              0
Rx Auth. Successes             0
Rx Auth. Failures              0

Switch(nas)# show status
Port Port State      Last Source      Last ID      QoS VLAN
-----
-----
1   Link Down        -                -
2   Link Down        -                -
3   Link Down        -                -
4   Link Down        -                -

```

33 PoE Commands

PoE is an acronym for Power Over Ethernet.

Power Over Ethernet is used to transmit electrical power, to remote devices over standard Ethernet cable. It could for example be used for powering IP telephones, wireless LAN access points and other equipment, where it would be difficult or expensive to connect the equipment to main power supply.



This feature only applies in some models with Power over Ethernet (PoE) feature. The models without PoE are not available to use this command.

Command	Function
max-power	Configure PoE maximum power per port
mode	Configure PoE mode
priority	Configure PoE priority
reset-port	Reset PoE port
retry-time	Configure the retry time of PoE port
show	Show PoE information

33.1 *delay-mode*

Using this command you can configure PoE Power Delay mode.

Syntax:

```
delay-mode <port-list> disable|enable
```

Parameter:

<port-list>

available value is from 1 to 24 format:1,3-5

disable

Disables PoE Power Delay

enable

Enables PoE Power Delay

Example:

```
Switch(poe)# delay-mode 3 enable
Switch(poe)#
```

33.2 *delay-time*

Using this command you can configure PoE Power Delay Time.

Syntax:

```
delay-time <port-list>
```

Parameter:

<port-list>

available value is from 1 to 24 format:1,3-5

<0-300>

Delay Time : 0 ~ 300(sec)

Example:

```
Switch(poe)# delay-time 5 60  
Switch(poe)#
```

33.3 *detection*

Using this command you can configure PoE port capacitor detection.

Syntax:

```
detection <port-list> 4-point|both|legacy
```

Parameter:

<port-list>

available value is from 1 to 24 format:1,3-5

4-point

IEEE 802.3af 4-point detection only

both

IEEE 802.3af 4-point detection followed by legacy

legacy

Legacy capacitive detection only

Example:

```
Switch(poe)# detection 5 legacy  
Switch(poe)#
```

33.4 *failure-action*

Using this command you can configure PoE Auto Check Failure Action.

Syntax:

```
failure-action <port-list> nothing|reboot-Remote-PD
```

Parameter:

<port-list>

available value is from 1 to 24 format:1,3-5

nothing

Failure Action : Nothing

reboot-Remote-PD

Failure Action : Reboot Remote PD

Example:

```
Switch(poe)# failure-action 5 reboot-Remote-PD
Switch(poe)#
```

33.5 *hour*

Using this command you can configure PoE Schedule Hour.

Syntax:

```
hour <port-list> <0-23> (All|Sun) (disable|enable)
```

Parameter:

<port-list>

available value is from 1 to 24 format:1,3-5

<0-23>

Hour : 0 ~ 23

All

Select all week

Sun

Sunday

disable

Disables PoE Schedule

enable

Enables PoE Schedule

Example:

```
Switch(poe)# hour 5 20 All enable
Switch(poe)#
```

33.6 *interval-time*

Using this command you can configure PoE Auto Check Interval Time.

Syntax:

```
interval-time <port-list> <10-120>
```

Parameter:**<port-list>**

available value is from 1 to 24 format:1,3-5

<10-120>

Interval Time : 10 ~ 120(sec)

Example:

```
Switch(poe)# interval-time 5 60
Switch(poe)#
```

33.7 *max-power*

Using this command you can configure PoE maximum power per port.

Syntax:

```
max-power <port-list> <port-power>
```

Parameter:**<port-list>:**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<port-power>

The Maximum Power value contains a numerical value that indicates the maximum power in watts that can be delivered to a remote device. The maximum allowed value is 30 W

Example:

```
Switch(poe)# max-power 1 30
Switch(poe)# max-power 2 28
Switch(poe)# show config
```

```

Primary Power Supply [W]      : 250
Retry Time (seconds)         : 60

Port  Mode      Priority  Max. Power [W]
-----
1     Enabled   Low      30.0
2     Enabled   Low      28.0
3     Enabled   Low      15.4

```

33.8 mode

Using this command you can configure PoE mode.

Syntax:

```
mode <port-list> disable|enable
```

Parameter:

<port-list>:

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables PoE operation.

enable

Enables PoE operation

Example:

```

Switch(poe)# mode 1 disable
Switch(poe)# show config
Primary Power Supply [W]      : 250
Retry Time (seconds)         : 60

Port  Mode      Priority  Max. Power [W]
-----
1     Disabled   Low      30.0
2     Enabled   Low      28.0
3     Enabled   Low      15.4

```

33.9 ping-check

Using this command you can enable or disable POE Ping Check.

Syntax:

```
ping-check disable|enable
```

Parameter:**disable**

Disables POE Ping Check

enable

Enables POE Ping Check

Example:

```
Switch(poe)# ping-check enable
Switch(poe)#
```

33.10 *ping-ip-addr*

Using this command you can configure PoE Ping IP Address.

Syntax:

```
ping-ip-addr <port-list> <ip-address>
```

Parameter:**<port-list>**

available value is from 1 to 24 format:1,3-5

<ip-address>

Set PoE Ping IP Address

Example:

```
Switch(poe)# ping-ip-addr 5 192.168.2.200
Switch(poe)#
```

33.11 *ping-retry-time*

Using this command you can configure PoE Auto Check Retry Time.

Syntax:

```
ping-retry-time <port-list>
```

Parameter:**<port-list>**

available value is from 1 to 24 format:1,3-5

<1-5>

Retry Time : 1 ~ 5

Example:

```
Switch(poe)# ping-retry-time 5 3
Switch(poe)#
```

33.12 *priority*

Using this command you can configure PoE priority.

Syntax:

```
priority <port-list> critical|high|low
```

Parameter:**<port-list>:**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

critical

Set priority to critical.

high

Set priority to high.

low

Set priority to low.

Example:

```
Switch(poe)# priority 1 critical
Switch(poe)# priority 2 high
Switch(poe)# priority 3 low
Switch(poe)# show config
Primary Power Supply [W]      : 250
Retry Time (seconds)          : 60

Port  Mode      Priority  Max. Power [W]
-----
1     Disabled   Critical  30.0
2     Enabled    High     28.0
3     Enabled    Low      15.4
```

33.13 *reboot-time*

Using this command you can configure PoE Auto Check Reboot Time.

Syntax:

```
reboot-time <3-120>
```

Parameter:**<port-list>**

available value is from 1 to 24 format:1,3-5

<3-120>

Reboot Time : 3 ~ 120(sec)

Example:

```
Switch(poe)# reboot-time 5 60
Switch(poe)#
```

33.14 *reset-port*

Using this command you can reset PoE port.

Syntax:

```
reset-port <port-list>
```

Parameter:**<port-list>:**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(poe)# reset-port 1
Switch(poe)# show config
Primary Power Supply [W]      : 250
Retry Time (seconds)          : 60

Port  Mode      Priority  Max. Power [W]
-----
1     Disabled   Critical  30.0
2     Enabled    High     28.0
3     Enabled    Low      15.4
```

33.15 *retry-time*

Using this command you can configure the retry time of PoE port.

Syntax:

```
retry-time disable|<retry-period>
```

Parameter:**disable**

Disables to try to turn on a overloaded PoE port.

<retry-period>

The period (in seconds) for trying to turn on a overloaded port. Available values are 5,10,20,30,40,50,60.

Example:

```
Switch(poe)# retry-time 40
Switch(poe)# show config
Primary Power Supply [W]      : 250
Retry Time (seconds)          : 40
```

33.16 *schedule-mode*

Using this command you can configure PoE Schedule mode.

Syntax:

```
schedule-mode <port-list> disable|enable
```

Parameter:**<port-list>**

available value is from 1 to 24 format:1,3-5

disable

Disables PoE Schedule

enable

Enables PoE Schedule

Example:

```
Switch(poe)# schedule-mode 5 enable
Switch(poe)#
```

33.17 *select-all*

Using this command you can configure PoE Schedule Select All.

Syntax:

```
select-all <port-list>
```

Parameter:**<port-list>**

available value is from 1 to 24 format:1,3-5

disable

Disables PoE Schedule Select All

enable

Enables PoE Schedule Select All

Example:

```
Switch(poe)# select-all 5 enable
Switch(poe)#
```

33.18 *show*

Using this command you can show the current PoE configuration and statistics.

Syntax:

```
show config|status
```

Parameter:**config**

Shows PoE configuration.

status

Shows PoE status.

Example:

```
Switch(poe)# show config
Primary Power Supply [W]      : 250
Retry Time (seconds)         : 60

Port  Mode      Priority  Max. Power [W]
-----
1     Enabled    Low      15.4
2     Enabled    Low      15.4
3     Enabled    Low      15.4

Switch(poe)# show status

      PD   Power   Power   Power   Current
Port  Class Requested Allocated Used     Used     Priority Port Status
-----
1     0     0.0    [W] 0.0    [W] 0.0    [W] 0    [mA] Low    No PD detected
2     0     0.0    [W] 0.0    [W] 0.0    [W] 0    [mA] Low    No PD detected
3     0     0.0    [W] 0.0    [W] 0.0    [W] 0    [mA] Low    No PD detected
```

34 Port configuration Commands

This chapter describes how to view the current port configuration and how to configure ports to non-default settings, including:

- Linkup/Linkdown
- Speed (Current and configured)
- Flow Control (Current Rx, Current Tx and Configured)
- Maximum Frame Size
- Excessive Collision Mode
- Power Control

Command	Function
clear	Clear port counter
description	Interface specific description
excessive-collision	Configure excessive collision operation
flow-control	Configure flow operation
max-frame	Configure maximum receive frame size
port-state	Configure port state operation
power-saving	Configure power saving operation
show	Show port information
speed-duplex	Configure speed duplex operation

34.1 *clear*

Using this command you can clear port counter.

Syntax:

```
clear <port-list>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(port)# clear 1
Switch(port)# show simple-counter
Port      Packets      Bytes      Errors      Drops
Filtered
-----
1/Rx      0                0          0          0
```

0				
1/Tx	0	0	0	0

2/Rx	0	0	0	0
0				
2/Tx	0	0	0	0

3/Rx	0	0	0	0
0				
3/Tx	0	0	0	0

34.2 description

Using this command you can define a specific description for each port.

Syntax:

```
description <port-list> <desc>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<desc>

Up to 47 characters describing this interface

Example:

```
Switch(port)# description 1 david
Switch(port)# show configuration
Port  State   Speed Duplex  Flow Control  Max. Frame  Excessive  Power
Description
-----
1      Enabled  Auto      -           10056        -          -
david
-----
2      Enabled  Auto      -           10056        -          -
-----
3      Enabled  Auto      -           10056        -          -
-----
```

34.3 excessive-collision

Using this command you can configure excessive collision operation for each port.

Syntax:

```
excessive-collision <port-list> discard|restart
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

discard

Discard the packet when excessive collision

restart

Retransmit the packet, regardless of the number of collisions

Example:

```
Switch(port)# excessive-collision 21 restart
Switch(port)# show configuration
Port State Speed Duplex Flow Control Max. Frame Excessive Power
Description
-----
1 Enabled Auto - 10056 - -
david
-----
2 Enabled Auto - 10056 - -
-----
21 Enabled SFP_Auto_AMS Disabled 10056 Restart Disabled
-----
22 Enabled SFP_Auto_AMS Disabled 10056 Discard Disabled
-----
```

34.4 flow-control

Using this command you can configure flow operation for each port.

Syntax:

```
flow-control <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables flow control operation on the selected port(s).

enable

Enables flow control operation on the selected port(s).

Example:

```
Switch(port)# flow-control 21 enable
Switch(port)# show configuration
Port State   Speed Duplex Flow Control Max. Frame Excessive Power
Description
-----
1    Enabled Auto      -           10056      -         -
david
-----
2    Enabled Auto      -           10056      -         -
-----
21   Enabled SFP_Auto_AMS Enabled     10056      Restart   Disabled
-----
22   Enabled SFP_Auto_AMS Disabled    10056      Discard   Disabled
-----
23   Enabled SFP_Auto_AMS Disabled    10056      Discard   Disabled
-----
```

34.5 *max-frame*

Using this command you can configure maximum receive frame size for each port.

Syntax:

```
max-frame <port-list> <1518-10056>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<1518-10056>

Maximum receive frame size in bytes

Example:

```
Switch(port)# max-frame 1 1600
Switch(port)# show configuration
Port State   Speed Duplex Flow Control Max. Frame Excessive Power
Description
-----
1    Enabled Auto      -           1600      -         -
david
-----
2    Enabled Auto      -           10056      -         -
-----
3    Enabled Auto      -           10056      -         -
-----
```

34.6 *port-state*

Using this command you can configure port state operation.

Syntax:

```
port-state <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables port state operation on the selected port(s).

enable

Enables port state operation on the selected port(s).

Example:

```
Switch(port)# port-state 1 disable
Switch(port)# show configuration
Port State   Speed Duplex Flow Control Max. Frame Excessive Power
Description
-----
1    Disabled Auto      -          1600      -      -
david
-----
2    Enabled  Auto      -          10056     -      -
-----
3    Enabled  Auto      -          10056     -      -
```

34.7 *power-saving*

Using this command you can configure power saving operation.

Syntax:

```
power-saving <port-list> actiphy|disable|dynamic|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

actiphy

Enables ActiPHY power control on the selected port(s).

disable

Disables power saving on the selected port(s).

dynamic

Enables dynamic power control on the selected port(s).

enable

Enables power saving on the selected port(s).

Example:

```
Switch(port)# power-saving 21 actiphy
Switch(port)# power-saving 22 dynamic
Switch(port)# power-saving 23 enable
Switch(port)# show configuration
```

Port	State	Speed	Duplex	Flow Control	Max. Frame	Excessive	Power
1	Disabled	Auto		-	1600	-	-
2	Enabled	Auto		-	10056	-	-
21	Enabled	SFP_Auto_AMS	Enabled		10056	Restart	ActiPHY
22	Enabled	SFP_Auto_AMS	Disabled		10056	Discard	Dynamic
23	Enabled	SFP_Auto_AMS	Disabled		10056	Discard	Enabled
24	Enabled	SFP_Auto_AMS	Disabled		10056	Discard	Disabled

34.8 show

Using this command you can show port information.

Syntax:

```
show configuration
show detail-counter <port>
show sfp <sfp-port>
show simple-counter
show status [<port-list>]
```

Parameter:**configuration**

Shows port configuration

detail-counter

Shows detailed traffic statistics for specific switch port

<port>

Port number

sfp

Shows sfp information

<sfp-port>

SFP port number

simple-counter

Shows general traffic statistics for all switch ports

status

Shows port status

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

Switch(port)# show simple-counter					
Port	Packets	Bytes	Errors	Drops	
Filtered					

1/Rx	0	0	0	0	0
0					
1/Tx	0	0	0	0	0

2/Rx	0	0	0	0	0
0					
2/Tx	0	0	0	0	0

3/Rx	0	0	0	0	0
0					
3/Tx	0	0	0	0	0

4/Rx	0	0	0	0	0
0					
4/Tx	0	0	0	0	0

21/Rx	37999	14338676	10258	6	
6					
21/Tx	8922	1817882	0	0	

22/Rx	0	0	0	0	0
0					
22/Tx	0	0	0	0	0

23/Rx	0	0	0	0	0
0					
23/Tx	0	0	0	0	0

24/Rx 3	10875	2276667	0	3
24/Tx	39016	14923782	0	0

34.9 *speed-duplex*

Using this command you can configure speed duplex operation.

Syntax:

```
speed-duplex <port-list> 10-full|10-half|...|disable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

10-full

Force speed duplex to 10-full operation

10-half

Force speed duplex to 10-half operation

100-full

Force speed duplex to 100-full operation

100-half

Force speed duplex to 100-half operation

1000-full

Force speed duplex to 1000-full operation

auto

Enables auto speed duplex configuration

disable

Disables the switch port operation

Example:

```
Switch(port)# speed-duplex 1 100-full
Switch(port)# speed-duplex 2 1000-full
Switch(port)# show configuration
Port State   Speed Duplex Flow Control Max. Frame Excessive Power
Description
-----
1   Disabled 100 Full      -           1600      -         -
david
-----
2   Enabled  1G Full      -          10056      -         -
-----
```

35 Port security Commands

This section shows you how to configure the Port Security settings of the Switch. You can use the Port Security feature to restrict input to an interface by limiting and identifying MAC addresses.

Command	Function
action	Configure the action involved with exceeding the limit
aging	Configure the aging mode and period
limit	Configure the max. number of MAC addresses that can be learned on the port
mode	Configure the global limit control mode
port-mode	Configure the port mode
reopen	Reopen one or more ports whose limit is exceeded and shut down
show	Show port security status

35.1 *action*

Using this command you can configure the action involved with exceeding the limit.

Syntax:

```
action <port-list> both|none|shutdown|trap
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

both

Send a SNMP trap and shutdown the port

none

Do nothing

shutdown

Shutdown the port

trap

Send a SNMP trap

Example:

```
Switch(port-security)# action 1 both
Switch(port-security)# action 2 none
Switch(port-security)# action 3 shutdown
Switch(port-security)# action 4 trap
Switch(port-security)# show config
Mode      : Disabled
```

```

Aging      : Disabled
Age Period: 3600

Port  Mode      Limit  Action
-----
1     Disabled    4     Trap & Shutdown
2     Disabled    4     None
3     Disabled    4     Shutdown
4     Disabled    4     Trap
5     Disabled    4     None

```

35.2 *aging*

Using this command you can configure the aging mode and period.

Syntax:

```

aging disable
aging enable <10-10000000>

```

Parameter:

disable

Disables aging.

enable

Enables aging.

<10-10000000>

Aging time in seconds between checks for activity on a MAC address

Example:

```

Switch(port-security)# aging enable 20
Switch(port-security)# show config
Mode      : Disabled
Aging     : Enabled
Age Period: 20

```

35.3 *limit*

Using this command you can configure the max. number of MAC addresses that can be learned on the port.

Syntax:

```

limit <port-list> <1-1024>

```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<1-1024>

Max. number of MAC addresses on selected port (s).

Example:

```
Switch(port-security)# limit 1 999
Switch(port-security)# show config
Mode      : Disabled
Aging     : Enabled
Age Period: 20
```

Port	Mode	Limit	Action
1	Disabled	999	Trap & Shutdown
2	Disabled	4	None
3	Disabled	4	Shutdown

35.4 mode

Using this command you can configure the global limit control mode.

Syntax:

```
mode disable|enable
```

Parameter:**disable**

Globally disable port security.

enable

Globally enable port security.

Example:

```
Switch(port-security)# mode enable
Switch(port-security)# show config
Mode      : Enabled
Aging     : Enabled
Age Period: 20
```

35.5 port-mode

Using this command you can configure the port mode.

Syntax:

```
port-mode <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables port security on selected port(s).

enable

Enables port security on selected port(s).

Example:

```
Switch(port-security)# port-mode 1 enable
Switch(port-security)# show config
Mode      : Disabled
Aging     : Enabled
Age Period: 20

Port  Mode      Limit  Action
----  -
1     Enabled    999   Trap & Shutdown
2     Disabled    4     None
3     Disabled    4     Shutdown
4     Disabled    4     Trap
5     Disabled    4     None
```

35.6 reopen

Using this command you can reopen one or more ports whose limit is exceeded and shut down.

Syntax:

```
reopen <port-list>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(port-security)# reopen 1
Switch(port-security)# show config
Mode      : Disabled
Aging     : Enabled
Age Period: 20

Port  Mode      Limit  Action
```

	Mode	Limit	Action
1	Enabled	999	Trap & Shutdown
2	Disabled	4	None
3	Disabled	4	Shutdown
4	Disabled	4	Trap

35.7 show

Using this command you can show port security status.

Syntax:

```
show config|switch-status
show port-status <port>
```

Parameter:

config

Shows port security configuration

port-status

Shows MAC addresses learned by port security

<port>

Port number, possible values depending on distinct hardware model.

switch-status

Shows port security switch status

Example:

```
Switch(port-security)# show config
Mode      : Disabled
Aging     : Disabled
Age Period: 3600

Port  Mode      Limit  Action
----  -
1     Disabled   4     None
2     Disabled   4     None
3     Disabled   4     None
4     Disabled   4     None

Switch(port-security)# show port-status 1
MAC Address      VID  State      Time of Addition      Age/Hold
Time
-----
<none>

Switch(port-security)# show switch-status
Users:
L = Limit Control
8 = 802.1X
D = DHCP Snooping
```

Port	Users	State	MAC Count
1	---	Disabled	0
2	---	Disabled	0
3	---	Disabled	0

36 Privilege level Commands

This section provides an overview of the privilege levels. The switch provides user set Account, Aggregation, Diagnostics, EEE, GARP, GVRP, IP, IPMC Snooping LACP LLDP LLDP MED MAC Table MRP MVR MVRP Maintenance Mirroring POE Ports Private VLANs QoS SMTP SNMP Security Spanning Tree System Trap Event VCL VLANs Voice VLAN Privilege Levels form 1 to 15.

Command	Function
group	Configure a privilege level group
show	Show privilege configuration

36.1 *group*

Using this command you can assign a privilege group to a certain privilege level.

Syntax:

```
group <group-name> <1-15>
```

Parameter:

<group-name>

Privilege group name

- Account
- Aggregation
- Diagnostics
- EEE
- Easyport
- GARP
- GVRP
- IP
- IPMC_Snooping
- LACP
- LLDP
- LLDP_MED
- Loop_Protect
- MAC_Table
- MVR
- Maintenance
- Mirroring
- PoE
- Ports
- Private_VLANs
- QoS
- SFlow

- SMTP
- SNMP
- Security
- Single_IP
- Spanning_Tree
- System
- Trap_Event
- UPnP
- VCL
- VLANs
- Voice_VLAN

<1-15>

Privilege level

Example:

```
Switch(privilege)# group account 13
Switch(privilege)# show
Privilege Current Level: 15

Group Name                      Privilege Level
-----
Account                         13
Aggregation                     10
Diagnostics                     10
```

36.2 *show*

Using this command you can show the current privilege group configuration.

Syntax:

show

Example:

```
Switch(privilege)# show
Privilege Current Level: 15

Group Name                      Privilege Level
-----
Account                         13
Aggregation                     10
Diagnostics                     10
EPS                             10
ERPS                             10
ETH_LINK_OAM                    10
EVC                             10
GARP                             10
GVRP                             10
IP                              10
IPMC_Snooping                   10
LACP                             10
```

LLDP	10
LLDP_MED	10
Loop_Protect	10
MAC_Table	10
MEP	10
MVR	10
Maintenance	15
Mirroring	10
PTP	10
Ports	10
Private_VLANS	10
QoS	10
SMTP	10
SNMP	10
Security	10
Spanning_Tree	10
System	10
Trap_Event	10
VCL	10
VLAN_Translation	10
VLANS	10

37 Private VLAN Commands

In a private VLAN, communication between ports in that private VLAN is not permitted. A VLAN can be configured as a private VLAN.

Command	Function
delete	Delete private VLAN group
port-isolate	Configure port isolation
private-vlan	Configure private VLAN group
show	Show private VLAN information

37.1 *delete*

Using this command you can delete private VLAN group.

Syntax:

```
delete private-vlan <1-X>
```

Parameter:

<1-X>

Private VLAN ID, possible values depending on distinct hardware model.

Example:

```
Switch(pvlan)# delete private-vlan 12
```

37.2 *port-isolate*

Using this command you can configure port isolation.

Syntax:

```
port-isolate <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables port isolation

enable

Enables port isolation

Example:

```
Switch(pvlan)# port-isolate 1 enable
Switch(pvlan)# show port-isolate
Port   Isolation
-----
1      Enabled
2      Disabled
3      Disabled
4      Disabled
```

37.3 *private-vlan*

Using this command you can configure private VLAN group.

Syntax:

```
private-vlan <1-X> <port-list>
```

Parameter:**<1-X>**

Private VLAN ID, possible values depending on distinct hardware model.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(pvlan)# private-vlan 2 10
Switch(pvlan)# show private-vlan
PVLAN ID  Ports
-----
1         1-26
2         10
```

37.4 *show*

Using this command you can show private VLAN information.

Syntax:

```
show port-isolate|private-vlan
```

Parameter:**port-isolate**

Shows port isolation information

private-vlan

Shows private VLAN membership information



The default Private VLAN includes all port members on the switch. Use 26-port switch as example.

Example:

```
Switch(pvlan)# show port-isolate
Port  Isolation
----  -
1     Disabled
2     Disabled
3     Disabled
4     Disabled
5     Disabled Switch(garp)#

Switch(pvlan)# show private-vlan
PVLAN ID  Ports
-----
1         1-26
```

38 QoS Commands

The switch support four QoS queues per port with strict or weighted fair queuing scheduling. It supports QoS Control Lists (QCL) for advance programmable QoS classification, based on IEEE 802.1p, Ethertype, VID, IPv4/IPv6 DSCP and UDP/TCP ports and ranges.

High flexibility in the classification of incoming frames to a QoS class. The QoS classification looks for information up to Layer 4, including IPv4 and IPv6 DSCP, IPv4 TCP/UDP port numbers, and user priority of tagged frames. This QoS classification mechanism is implemented in a QoS control list (QCL). The QoS class assigned to a frame is used throughout the device for providing queuing, scheduling, and congestion control guarantees to the frame according to what was configured for that specific QoS class.

The switch support advanced memory control mechanisms providing excellent performance of all QoS classes under any traffic scenario, including jumbo frame. A super priority queue with dedicated memory and strict highest priority in the arbitration. The ingress super priority queue allows traffic recognized as CPU traffic to be received and queued for transmission to the CPU even when all the QoS class queues are congested.

Command	Function
delete	Delete QCE
dscp-classification	Configure DSCP ingress classification
dscp-map	Configure DSCP mapping table. This table is used to map QoS class and DP level based on DSCP value. DSCP value used to map QoS class and DPL is either translated DSCP value or incoming frame DSCP value
dscp-remap	Configure DSCP egress remap table. This table is used if the port egress remarking mode is 'remap' and the purpose is to map the DSCP and DP level to a new DSCP value
dscp-translation	Configure global ingress DSCP translation table. If port DSCP translation is enabled, translation table is used to translate incoming frame's DSCP value and translated value is used to map QoS class and DP level
dscp-trust	Configure trusted DSCP value which is used for QoS classification. The DSCP value to be checked for trust is either translated value if DSCP translation is enabled for the ingress port or incoming frame DSCP value if translation is disabled for the port. Trusted DSCP value is only used for QoS classification
port-classify	QoS ingress port classification
port-dscp	QoS port DSCP configuration
port-policer	Port policer
port-scheduler	QoS egress port schedulers
port-shaper	Port shaper
qce	Add or modify QoS control entry
queue-shaper	Queue shaper
show	Show QoS information
storm	Configure storm rate control
tag-remarking	QoS egress port tag remarking
show	Show the GVRP configuration

38.1 *delete*

Using this command you can delete QCE.

 If you set the GVRP on port then you could show the port GVRP statistics information or clear all record on port.

Syntax:

```
delete <1-256>
```

Parameter:

<1-256>

QCE ID must be exist

Example:

```
Switch(qos)# delete 1
```

38.2 *dscp-classification*

Using this command you can configure DSCP ingress classification.

Syntax:

```
dscp-classification map <class-list> <dpl-list> <0-63>  
dscp-classification mode <dscp-list> disable|enable
```

Parameter:

map

Configure DSCP ingress classification mapping table. This table is used to map DSCP from QoS class and DP level. The DSCP which needs to be classified depends on port DSCP classification and DSCP classification mode. Incoming frame DSCP may be translated before using the value for classification

<class-list>

QoS class list, available value is from 0 to 7

<dpl-list>

Drop precedence level list, available value is from 0 to 1

<0-63>

Mapped DSCP

mode

Configure DSCP ingress classification mode. If port DSCP classification is 'selected', DSCP will be classified based on QoS class and DP level only for DSCP value with classification mode 'enabled'. DSCP may be translated DSCP if translation is enabled for the port

<dscp-list>

DSCP list, format : 1,3,5-7

disable

Disables DSCP ingress classification

enable

Enables DSCP ingress classification

Example:

```
Switch(qos)# dscp-classification map 7 10
Switch(qos)# show class-map
QoS Class  DSCP
-----  -
0          0
1          0
2          0
3          0
4          0
5          0
6          0
7          10

Switch(qos)# dscp-classification mode 1 enable
Switch(qos)# show dscp-translation
      Ingress      Ingress      Egress
DSCP  Translation  Classify  Remap
-----  -
0      0           Disabled  0
1      1           Enabled   1
2      2           Disabled  2
3      3           Disabled  3
4      4           Disabled  4
```

38.3 dscp-map

Using this command you can configure DSCP mapping table.

Syntax:*dscp-map* <dscp-list> <0-7> <0-1>**Parameter:****<dscp-list>**

DSCP list, format : 1,3,5-7

<0-7>

QoS classenable The parameter let you enable GVRP function on port.

<0-1>

Drop Precedence Level

Example:

```
Switch(qos)# dscp-map 2 6 2
Switch(qos)# show dscp-map
DSCP    Trust      QoS Class  DP Level
-----
0  (BE)    Disabled  0         0
1         Disabled  0         0
2         Disabled  6         2
3         Disabled  0         0
```

38.4 *dscp-remap*

Using this command you can configure DSCP egress remap table.

Syntax:

```
dscp-remap <dscp-list> <dpl-list> <0-63>
```

Parameter:**<dscp-list>**

DSCP list, format : 1,3,5-7

<dpl-list>

Drop precedence level list, available value is from 0 to 1

<0-63>

Egress remapped DSCP

Example:

```
Switch(qos)# dscp-remap 3 44
Switch(qos)# show dscp-translation
      Ingress      Ingress      Egress
DSCP  Translation  Classify    Remap
-----
0      0           Disabled    0
1      1           Enabled     1
2      2           Disabled    2
3      3           Disabled    44
4      4           Disabled    4
```

38.5 *dscp-translation*

Using this command you can configure global ingress DSCP translation table.

Syntax:

```
dscp-translation <dscp-list> <0-63>
```

Parameter:**<dscp-list>**

DSCP list, format : 1,3,5-7

<0-63>

Translated DSCP

Example:

```
Switch(qos)# dscp-translation 4 55
Switch(qos)# show dscp-translation
```

DSCP	Ingress Translation	Ingress Classify	Egress Remap
0	0	Disabled	0
1	1	Enabled	1
2	2	Disabled	2
3	3	Disabled	44
4	55	Disabled	4
5	5	Disabled	5

38.6 dscp-trust

Using this command you can configure trusted DSCP value which is used for QoS classification.

Syntax:

```
dscp-trust <dscp-list> disable|enable
```

Parameter:**<dscp-list>**

DSCP list, format : 1,3,5-7

disable

Set DSCP as untrusted DSCP

enable

Set DSCP as trusted DSCP

Example:

```
Switch(qos)# dscp-trust 6 enable
Switch(qos)# show dscp-map
```

DSCP	Trust	QoS Class	DP Level
0	(BE)	Disabled	0
1		Disabled	0
2		Disabled	6
3		Disabled	0
4		Disabled	0
5		Disabled	0
6		Enabled	0
7		Disabled	0

38.7 *port-classify*

Using this command you can configure QoS ingress port classification.

Syntax:

```
port-classify class <port-list> <0-7>
port-classify dei <port-list> <0-1>
port-classify dpl <port-list> <0-1>
port-classify dscp <port-list> disable|enable
port-classify map <port-list> <0-7> <0-1> <0-7> <0-1>
port-classify pcpc <port-list> <0-7>
port-classify tag <port-list> disable|enable
```

Parameter:

class

Configure the default QoS class

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-7>

QoS class for frames not classified in any other way. There is a one to one mapping between QoS class, queue and priority. A QoS class of 0 (zero) has the lowest priority

dei

Configure the default DEI for untagged frames

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-1>

Drop Eligible Indicator. It is a 1-bit field in the VLAN tag

dpl

Configure the default DP level

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-1>

DP level for frames not classified in any other way

dscp

Configure DSCP based classification mode

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables DSCP based classification

enable

Enables DSCP based classification

map

Configure the port classification map. This map is used when port classification tag is enabled, and the purpose is to translate the Priority Code Point (PCP) and Drop Eligible Indicator (DEI) from a tagged frame to QoS class and DP level

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-7>

Priority Code Point

<0-1>

Drop Eligible Indicator

<0-7>

QoS class

<0-1>

Drop precedence level

pcp

Configure the default PCP for untagged frames

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-7>

Priority Code Point. It is a 3-bit field storing the priority level for the 802.1Q frame

tag

Configure the classification mode for tagged frames

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Use default QoS class and DP level for tagged frames

enable

Use mapped versions of PCP and DEI for tagged frames

Example:

```
Switch(qos)# port-classify class 7 4
Switch(qos)# show port-classify
Port  QoS class  DP level  PCP  DEI  Tag class.  DSCP Based
-----
```

```

1      0      0      0      0      Disabled      Disabled
2      0      0      0      0      Disabled      Disabled
3      0      0      0      0      Disabled      Disabled
4      0      0      0      0      Disabled      Disabled
5      0      0      0      0      Disabled      Disabled
6      0      0      0      0      Disabled      Disabled
7      4      0      0      0      Disabled      Disabled
8      0      0      0      0      Disabled      Disabled

```

```
Switch(qos)# port-classify dei 1 1
```

```
Switch(qos)# show port-classify
```

Port	QoS class	DP level	PCP	DEI	Tag class.	DSCP Based
1	0	0	0	1	Disabled	Disabled
2	0	0	0	0	Disabled	Disabled
3	0	0	0	0	Disabled	Disabled

```
Switch(qos)# port-classify dpl 2 3
```

```
Switch(qos)# show port-classify
```

Port	QoS class	DP level	PCP	DEI	Tag class.	DSCP Based
1	0	0	0	1	Disabled	Disabled
2	0	3	0	0	Disabled	Disabled
3	0	0	0	0	Disabled	Disabled

```
Switch(qos)# port-classify dscp 3 enable
```

```
Switch(qos)# show port-classify
```

Port	QoS class	DP level	PCP	DEI	Tag class.	DSCP Based
1	0	0	0	1	Disabled	Disabled
2	0	3	0	0	Disabled	Disabled
3	0	0	0	0	Disabled	Enabled

```
Switch(qos)# port-classify map 4 5 1 6 3
```

```
Switch(qos)# show port-map 4
```

Port	PCP	DEI	QoS class	DP level
4	0	0	1	0
	0	1	1	1
	1	0	0	0
	1	1	0	1
	2	0	2	0
	2	1	2	1
	3	0	3	0
	3	1	3	1
	4	0	4	0
	4	1	4	1
	5	0	5	0
	5	1	6	3
	6	0	6	0
	6	1	6	1
	7	0	7	0
	7	1	7	1

```
Switch(qos)# port-classify pcps 5 3
```

```
Switch(qos)# show port-classify
```

Port	QoS class	DP level	PCP	DEI	Tag class.	DSCP Based
1	0	0	0	1	Disabled	Disabled
2	0	3	0	0	Disabled	Disabled
3	0	0	0	0	Disabled	Enabled

```

4      0      0      0      0      Disabled      Disabled
5      0      0      3      0      Disabled      Disabled

```

```
Switch(qos)# port-classify tag 6 enable
```

```
Switch(qos)# show port-classify
```

Port	QoS class	DP level	PCP	DEI	Tag class.	DSCP Based
1	0	0	0	1	Disabled	Disabled
2	0	3	0	0	Disabled	Disabled
3	0	0	0	0	Disabled	Enabled
4	0	0	0	0	Disabled	Disabled
5	0	0	3	0	Disabled	Disabled
6	0	0	0	0	Enabled	Disabled

38.8 port-dscp

Using this command you can do QoS port DSCP configuration.

Syntax:

```

port-dscp classification <port-list> all|disable|selected|zero
port-dscp egress-remark <port-list> (disable|enable) (remap_dp_aware|remap_dp_unaware)
port-dscp translation <port-list> disable|enable

```

Parameter:

classification

Configure DSCP classification based on QoS class and DP level. This enables per port to map new DSCP value based on QoS class and DP level

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

all

Classify all DSCP

disable

Disables DSCP ingress classification

selected

Classify only selected DSCP for which classification is enabled as specified in DSCP Translation window for the specific DSCP

zero

Classify DSCP if DSCP = 0

egress-remark

Configure the port DSCP remarking mode

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables DSCP egress rewrite

enable

Enables DSCP egress rewrite with the value received from analyzer

remap_dp_aware

Rewrite DSCP in egress frame with remapped DSCP where remap is DP aware and DP = 1

remap_dp_unaware

Rewrite DSCP in egress frame with remapped DSCP where remap is DP unaware or DP = 0

translation

Configure DSCP ingress translation mode. If translation is enabled for a port, incoming frame DSCP value is translated and translated value is used for QoS classification

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables DSCP ingress translation

enable

Enables DSCP ingress translation

Example:

```
Switch(qos)# port-dscp classification 1 all
Switch(qos)# port-dscp classification 2 selected
Switch(qos)# port-dscp classification 3 zero
Switch(qos)# show port-dscp
Port  DSCP translation  Ingress classification  Egress remark
----  -
1     Disabled         All                     Disabled
2     Disabled         Selected               Disabled
3     Disabled         DSCP = 0              Disabled
4     Disabled         Disabled               Disabled

Switch(qos)# port-dscp egress-remark 4 enable
Switch(qos)# port-dscp egress-remark 5 remap
Switch(qos)# show port-dscp
Port  DSCP translation  Ingress classification  Egress remark
----  -
1     Disabled         All                     Disabled
2     Disabled         Selected               Disabled
3     Disabled         DSCP = 0              Disabled
4     Disabled         Disabled               Enabled
5     Disabled         Disabled               Remapped

Switch(qos)# port-dscp translation 6 enable
Switch(qos)# show port-dscp
Port  DSCP translation  Ingress classification  Egress remark
----  -
1     Disabled         All                     Disabled
2     Disabled         Selected               Disabled
3     Disabled         DSCP = 0              Disabled
```

4	Disabled	Disabled	Enabled
5	Disabled	Disabled	Remapped
6	Enabled	Disabled	Disabled

38.9 *port-policer*

Using this command you can do Port policer.

Syntax:

```
port-policer flow-control|mode <port-list> disable|enable
port-policer rate <port-list> Kbps|Kfps|Mbps|fps <rate-range>
```

Parameter:

flow-control

Configure the port policer flow control mode

mode

Configure the port policer mode

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables port policer flow control

enable

Enables port policer flow control

rate

Configure the port policer rate

Kbps

Rate in kilo bits per second (Kbps)

Kfps

Rate in kilo frame per second (Kfps)

Mbps

Rate in mega bits per second (Mbps)

fps

Rate in frame per second (fps)

<rate-range>

<100-10000000>

Rate for Kbps

<1-10000>

Rate for Kfps

<1-10000>

Rate for Mbps

<100-10000000>

Rate for fps

Example:

```
Switch(qos)# port-policer flow-control 1 enable
```

```
Switch(qos)# show port-policer
```

Port	Mode	Rate	Flow Control
1	Disabled	500 kbps	Enabled
2	Disabled	500 kbps	Disabled
3	Disabled	500 kbps	Disabled
4	Disabled	500 kbps	Disabled

```
Switch(qos)# port-policer mode 2 enable
```

```
Switch(qos)# show port-policer
```

Port	Mode	Rate	Flow Control
1	Disabled	500 kbps	Disabled
2	Enabled	500 kbps	Disabled
3	Disabled	500 kbps	Disabled
4	Disabled	500 kbps	Disabled

```
Switch(qos)# port-policer rate 3 mbps 99
```

```
Switch(qos)# show port-policer
```

Port	Mode	Rate	Flow Control
1	Disabled	500 kbps	Disabled
2	Disabled	500 kbps	Disabled
3	Disabled	99 Mbps	Disabled
4	Disabled	500 kbps	Disabled

38.10 *port-scheduler*

Using this command you can do QoS egress port schedulers.

Syntax:

```
port-scheduler mode <port-list> strict|weighted
port-scheduler weight <port-list> <0-5> <1-100>
```

Parameter:**mode**

Configure the port scheduler mode

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

strict

Strict priority scheduler mode

weighted

Weighted scheduler mode

weight

Configure the port scheduler weight

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-5>

Weighted queue

<1-100>

Scheduler weight

Example:

```
Switch(qos)# port-scheduler mode 1 weighted
Switch(qos)# show scheduler-mode
Port  Mode
----  -
1     Weighted
2     Strict
3     Strict

Switch(qos)# port-scheduler weight 2 5 99
Switch(qos)# show scheduler-weight 2
Port  Queue  Weight
----  -
2     0      17 (9%)
      1      17 (9%)
      2      17 (9%)
      3      17 (9%)
      4      17 (9%)
      5      99 (55%)
```

38.11 *port-shaper*

Using this command you can do Port shaper .

Syntax:

```
port-shaper mode <port-list> disable|enable
port-shaper rate <port-list> <100-10000000>
```

Parameter:**mode**

Configure the port shaper mode

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disable

enable

Enable

rate

Configure the port shaper rate

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<100-10000000>

Rate in kilo bits per second (Kbps)

Example:

```
Switch(qos)# port-shaper mode 1 enable
Switch(qos)# show port-shaper
Port  Mode      Rate
----  -
1     Enabled    500 kbps
2     Disabled   500 kbps
3     Disabled   500 kbps
```

```
Switch(qos)# port-shaper rate 2 999
Switch(qos)# show port-shaper
Port  Mode      Rate
----  -
1     Enabled    500 kbps
2     Disabled   999 kbps
3     Disabled   500 kbps
4     Disabled   500 kbps
```

38.12 *qce*

Using this command you can add or modify QoS control entry.

Syntax:

```
qce <1-256> <0-256> <port-list> any|etype|ipv4|ipv6|llc|snap
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**<1-256>**

If the QCE ID parameter <qce_id> is specified and an entry with this QCE ID already exists, the QCE will be modified. Otherwise, a new QCE will be added

<0-256>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<port-list>

Port member for QCE

any

Any frame type can match this QCE

etype

Only Ethernet Type frames can match this QCE

ipv4

Only IPv4 frames can match this QCE

ipv6

Only IPv6 frames can match this QCE

llc

Only LLC frames can match this QCE

snap

Only SNAP frames can match this QCE

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# class 7
Switch(qos/qce-etype)# classified-dscp 63
Switch(qos/qce-etype)# dei 1
Switch(qos/qce-etype)# dmac unicast
Switch(qos/qce-etype)# dp 3
Switch(qos/qce-etype)# pcp 5
Switch(qos/qce-etype)# smac any
Switch(qos/qce-etype)# tag enable
Switch(qos/qce-etype)# vid 21-25
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged          SMAC      : Any
VID      : 21-25          DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
Ether Type : Any          Class     : 7
                          DP       : 3
                          DSCP     : 63
```

38.12.1 *class*

Using this command you can configure action of QoS class for this QCE.

Syntax:

```
class default | <0-7>
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

class

Action of QoS class for this QCE

default

Basic classification

<0-7>

QoS class value

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# class 7
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged           SMAC      : Any
VID      : 21-25           DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
Ether Type : Any         Class     : 7
                          DP      : 3
                          DSCP    : 63
```

38.12.2 *classified-dscp*

Using this command you can configure action of DSCP for this QCE.

Syntax:

```
classified-dscp default | <0-63>
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

default

Basic classification

<0-63>

DSCP value

Example:

```

Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# classified-dscp 63
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged           SMAC      : Any
VID      : 21-25           DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
Ether Type : Any         Class     : 7
                          DP      : 3
                          DSCP    : 63

```

38.12.3 *dei*

Using this command you can configure whether frames can hit the action according to DEI.

Syntax:

dei any | <0-1>

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**any**

Don't care

<0-1>

Drop Eligible Indicator value

Example:

```

Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# dei 1
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged           SMAC      : Any
VID      : 21-25           DMAC Type: Unicast
PCP      : 5
DEI      : 1

```

Ethernet Parameters	Action Parameters
-----	-----
Ether Type : Any	Class : 7
	DP : 3
	DSCP : 63

38.12.4 *dmac*

Using this command you can configure destination MAC address for this QCE.

Syntax:

```
dmac any|broadcast|multicast|unicast
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

any

Don't care

broadcast

Frame must be broadcast

multicast

Frame must be multicast

unicast

Frame must be unicast

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# dmac unicast
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged          SMAC      : Any
VID      : 21-25          DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
Ether Type : Any          Class     : 7
                          DP      : 3
                          DSCP    : 63
```

38.12.5 *dp*

Using this command you can configure the action of drop precedence level for this QCE.

Syntax:

```
dp                default | <0-1>
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**default**

Basic classification

<0-1>

Drop precedence level

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# dp 3
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged          SMAC      : Any
VID      : 21-25          DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
Ether Type : Any          Class    : 7
                          DP      : 3
                          DSCP    : 63
```

38.12.6 end

Using this command you can finish QCE settings and return to QoS mode.

Syntax:

```
end
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# dscp any
Switch(qos/qce-etype)# end
Switch(qos)#
```

38.12.7 pcp

Using this command you can configure whether frames can hit the action according to PCP.

Syntax:

```
pcp 0-1|0-3|2-3|4-5|4-7|6-7|any|<0-7>
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**0-1**

Priority Code Point (0-1)

0-3

Priority Code Point (0-3)

2-3

Priority Code Point (2-3)

4-5

Priority Code Point (4-5)

4-7

Priority Code Point (4-7)

6-7

Priority Code Point (6-7)

any

Don't care

<0-7>

Priority Code Point

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# pcp 5
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged          SMAC      : Any
VID      : 21-25          DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
Ether Type : Any          Class     : 7
                          DP      : 3
                          DSCP    : 63
```

38.12.8 show

Using this command you can display current QCE configuration.

Syntax:

```
show
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# dscp any
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged          SMAC      : Any
VID      : 21-25          DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
Ether Type : Any          Class     : 7
                          DP       : 3
                          DSCP     : 63
```

38.12.9 *smac*

Using this command you can configure source MAC address for this QCE.

Syntax:

```
smac                <oui-address>|any
```

 The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

<oui-address>

A frame that hits this QCE matches this source OUI address value

any

Don't care

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# smac any
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged          SMAC      : Any
VID      : 21-25          DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
```

Ether Type : Any	Class : 7
	DP : 3
	DSCP : 63

38.12.10 *tag*

Using this command you can configure whether frames can hit the action according to the 802.1Q tagged.

Syntax:

```
tag any|disable|enable
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:

any

Don't care.

disable

Untagged frame only.

enable

Tagged frame only.

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# tag enable
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged          SMAC      : Any
VID      : 21-25          DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
Ether Type : Any          Class     : 7
                          DP       : 3
                          DSCP     : 63
```

38.12.11 *vid*

Using this command you can configure the VLAN ID filter for this QCE.

Syntax:

```
vid any|<vlan-range>
```



The availability of 2nd level parameters depends on the frame-type you select (any, ipv4, etc.).

Parameter:**any**

No VLAN ID filter is specified. (VLAN ID filter status is don't-care.)

<vlan-range>

A frame that hits this QCE matches this VLAN range

Example:

```
Switch(qos)# qce 13 23 25 etype
Switch(qos/qce-etype)# vid 21-25
Switch(qos/qce-etype)# show
QCE ID      : 13
Frame Type  : Ethernet          Port      : 25,29

VLAN Parameters          MAC Parameters
-----
Tag      : Tagged           SMAC      : Any
VID      : 21-25           DMAC Type: Unicast
PCP      : 5
DEI      : 1

Ethernet Parameters      Action Parameters
-----
Ether Type : Any         Class     : 7
                          DP      : 3
                          DSCP    : 63
```

38.13 *queue-shaper*

Using this command you can do Queue shaper.

Syntax:

```
queue-shaper excess|mode <port-list> <queue-list> disable|enable
queue-shaper rate <port-list> <queue-list> <100-1000000>
```

Parameter:**excess**

Configure the port queue excess bandwidth mode

mode

Configure the port queue shaper mode

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<queue-list>

Queue list, available value is from 0 to 7

disable

Disables use of excess bandwidth

enable

Enables use of excess bandwidth

rate

Configure the port queue shaper rate

<100-1000000>

Rate in kilo bits per second (Kbps)

Example:

```
Switch(qos)# queue-shaper excess 1 7 enable
Switch(qos)# show queue-shaper 1
```

Port	Queue	Mode	Rate	Excess
1	0	Disabled	500 kbps	Disabled
	1	Disabled	500 kbps	Disabled
	2	Disabled	500 kbps	Disabled
	3	Disabled	500 kbps	Disabled
	4	Disabled	500 kbps	Disabled
	5	Disabled	500 kbps	Disabled
	6	Disabled	500 kbps	Disabled
	7	Disabled	500 kbps	Enabled

38.14 show

Using this command you can show QoS information .

Syntax:

```
show (class-map|dscp-map|dscp-translation|port-classify|port-dscp|
    port-policer|port-shaper|scheduler-mode|storm|tag-remarking)

show (port-map|queue-shaper|remarking-map|scheduler-weight) <port-list>

show qce <1-256>
show qcl-status (combined|conflicts|static|voice-vlan)
```

Parameter:**class-map**

Shows QoS class and DP level to DSCP mapping

dscp-map

Shows DSCP to QoS class and DP level mapping

dscp-translation

Shows DSCP ingress and egress translation

port-classify

Shows QoS ingress port classification

port-dscp

Shows port DSCP configuration

port-policer

Shows port policer configuration

port-shaper

Shows port shaper configuration

scheduler-mode

Shows port scheduler mode configuration

storm

Shows storm control configuration

tag-remarking

Shows port tag remarking configuration

port-map

Shows port classification (PCP, DEI) to (QoS class, DP level) mapping table

queue-shaper

Shows port queue shaper configuration

remarking-map

Shows port tag remarking mapping table

scheduler-weight

Shows port scheduler weight configuration

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

qce

Shows QCL control list

<1-256>

QCE ID

qcl-status

Shows QCL status

combined

Shows the combined status

conflicts

Shows all conflict status

static

Shows the static user configured status

voice-vlan

Shows the status by Voice VLAN

Example:

```
Switch(qos)# show class-map
QoS Class  DSCP
```

```

-----
0          0
1          0
2          0
3          0
4          0
5          0
6          0
7          0

```

Switch(qos)# show dscp-map

DSCP	Trust	QoS Class	DP Level
0	(BE)	Disabled	0
1		Disabled	0
2		Disabled	0
3		Disabled	0

```

-----
0 (BE) Disabled 0 0
1 Disabled 0 0
2 Disabled 0 0
3 Disabled 0 0

```

Switch(qos)# show dscp-translation

DSCP	Ingress Translation	Ingress Classify	Egress Remap
0	0	Disabled	0
1	1	Disabled	1
2	2	Disabled	2
3	3	Disabled	3

```

-----
0 0 Disabled 0
1 1 Disabled 1
2 2 Disabled 2
3 3 Disabled 3

```

Switch(qos)# show port-classify

Port	QoS class	DP level	PCP	DEI	Tag class.	DSCP Based
1	0	0	0	0	Disabled	Disabled
2	0	0	0	0	Disabled	Disabled
3	0	0	0	0	Disabled	Disabled

Switch(qos)# show port-dscp

Port	DSCP translation	Ingress classification	Egress remark
1	Disabled	Disabled	Disabled
2	Disabled	Disabled	Disabled
3	Disabled	Disabled	Disabled

Switch(qos)# show port-map 1

Port	PCP	DEI	QoS class	DP level
1	0	0	1	0
	0	1	1	1
	1	0	0	0

Switch(qos)# show port-policer

Port	Mode	Rate	Flow Control
1	Disabled	500 kbps	Disabled
2	Disabled	500 kbps	Disabled
3	Disabled	500 kbps	Disabled

Switch(qos)# show port-shaper

Port	Mode	Rate
------	------	------

```

-----
1      Disabled    500 kbps
2      Disabled    500 kbps
3      Disabled    500 kbps

```

```
Switch(qos)# show qce 200
```

```

Switch(qos)# show qcl-status combined
Number of QCEs: 0
Switch(qos)# show qcl-status conflicts
Number of QCEs: 0
Switch(qos)# show qcl-status static
Number of QCEs: 0
Switch(qos)# show qcl-status voice-vlan

```

```
Switch(qos)# show queue-shaper 1
```

Port	Queue	Mode	Rate	Excess
1	0	Disabled	500 kbps	Disabled
	1	Disabled	500 kbps	Disabled
	2	Disabled	500 kbps	Disabled
	3	Disabled	500 kbps	Disabled
	4	Disabled	500 kbps	Disabled
	5	Disabled	500 kbps	Disabled
	6	Disabled	500 kbps	Disabled
	7	Disabled	500 kbps	Disabled

```
Switch(qos)# show remarking-map 1
```

Port	QoS class	DP level	PCP	DEI
1	0	0	1	0
	0	1	1	1
	1	0	0	0
	1	1	0	1
	2	0	2	0
	2	1	2	1
	3	0	3	0
	3	1	3	1
	4	0	4	0
	4	1	4	1
	5	0	5	0
	5	1	5	1
	6	0	6	0
	6	1	6	1
	7	0	7	0
	7	1	7	1

```
Switch(qos)# show scheduler-mode
```

Port	Mode
1	Strict
2	Strict
3	Strict

```
Switch(qos)# show scheduler-weight 1
```

Port	Queue	Weight
1	0	17 (17%)
	1	17 (17%)
	2	17 (17%)

```

3      17  (17%)
4      17  (17%)
5      17  (17%)

Switch(qos)# show storm
          Unicast          Broadcast          Unknown
Port  Mode      Rate      Mode      Rate      Mode      Rate
-----
1     Disabled  500 kbps Disabled  500 kbps Disabled  500 kbps
2     Disabled  500 kbps Disabled  500 kbps Disabled  500 kbps
3     Disabled  500 kbps Disabled  500 kbps Disabled  500 kbps
4     Disabled  500 kbps Disabled  500 kbps Disabled  500 kbps

Switch(qos)# show tag-remarking ?
<cr>
Switch(qos)# show tag-remarking
Port  Mode      PCP  DEI
-----
1     Classified  0    0
2     Classified  0    0
3     Classified  0    0

Switch(qos)# show wred
Queue  Mode      Min. Threshold  Max. DP 1  Max. DP 2  Max. DP 3
-----
0     Disabled  0                1          5          10
1     Disabled  0                1          5          10
2     Disabled  0                1          5          10
3     Disabled  0                1          5          10
4     Disabled  0                1          5          10
5     Disabled  0                1          5          10

```

38.15 *storm*

Using this command you can configure storm rate control.

Syntax:

```
storm (broadcast|multicast|unicast) (disable|enable)
```

Parameter:

broadcast

Broadcast frame storm control

unicast

Unicast frame storm control

unknown

Unknown frame storm control

disable

Disables port storm control

enable

Enables port storm control

Example:

```
Switch(qos)# show storm
Frame Type   State      Rate
-----
Unicast      Disabled    1 pps
Multicast    Disabled    1 pps
Broadcast    Disabled    1 pps
```

38.16 tag-remarking

Using this command you can do QoS egress port tag remarking.

Syntax:

```
tag-remarking dei <port-list> <0-1>
tag-remarking map <port-list> <class-list> <dpl-list> <0-7> <0-1>
tag-remarking mode <port-list> classified|default|mapped
tag-remarking pcp <port-list> <0-7>
```

Parameter:**dei**

Configure the default DEI. This value is used when port tag remarking mode is set to 'default'.

map

Configure the port tag remarking map. This map is used when port tag remarking mode is set to 'mapped', and the purpose is to translate the classified QoS class (0-7) and DP level (0-1) to PCP and DEI

mode

Configure the port tag remarking mode

pcp

Configure the default PCP. This value is used when port tag remarking mode is set to 'default'

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<class-list>

QoS class list, available value is from 0 to 7

<dpl-list>

Drop precedence level list, available value is from 0 to 1

<0-7>

Priority Code Point

<0-1>

Drop Eligible Indicator

classified

Use classified PCP/DEI values

default

Use default PCP/DEI values

mapped

Use mapped versions of QoS class and DP level

Example:

```
Switch(qos)# tag-remarking dei 1 1
Switch(qos)# tag-remarking mode 2 mapped
Switch(qos)# tag-remarking pcp 3 7
Switch(qos)# show tag-remarking
Port  Mode          PCP  DEI
----  -
1     Classified      0    1
2     Mapped          0    0
3     Classified      7    0
4     Classified      0    0

Switch(qos)# tag-remarking map 2 7 1 7 1
```

39 Reboot Commands

This section describes how to restart switch for any maintenance needs. Any configuration files or scripts that you saved in the switch should still be available afterwards.

Command	Function
reboot	Reboot the system

39.1 *reboot*

Using this command you can reboot the system.

Syntax:

```
reboot
```

Example:

```
Switch# reboot
```

40 SFlow Commands

The sFlow Collector configuration for the switch can be monitored and modified here. Up to 1 Collector is supported. This section allows for configuring sFlow collector IP type, sFlow collector IP Address,Port Number, for each sFlow Collector.

Command	Function
collector	sFlow Collector Configuration
sampler	sFlow sampler Configuration
show	Show sFlow

40.1 *collector*

Using this command you can set sFlow Collector Configuration.

Syntax:

```
collector IPv4|IPv6 <ip-address> <1-65535> <0-2147483647> <200-1500>
```

Parameter:

IPv4

IP type

IPv6

IP type

<ip-address>

IP address

<1-65535>

TCP/UDP port number. By default, the port number is 6343

<0-2147483647>

Set the receiver timeout for list of receiver ID (RID). Collector cannot collect samples unless receive timeout is not set.

<200-1500>

Set the receiver datagram length for list of receiver ID (RID).

Example:

```
Switch(sflow)# collector ipv4 192.168.100.100 6345 99 1500
Switch(sflow)# show
% Incomplete command
Switch(sflow)# show collector
      Configured      Current
-----
Collector Id    1          1
IP Type        IPv4        IPv4
```

IP Address	192.168.100.100	192.168.100.100
Port	6345	6345
Time Out	99	90 Timer is still alive!
Datagram Size	1500	1500

40.2 *sampler*

Using this command you can sFlow sampler Configuration .

Syntax:

```
sampler <port-list> ALL|RX|TX|none <0-4095> <14-200> <0-3600>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

ALL

Sample on both RX and TX

RX

Sample on RX

TX

Sample on TX

none

Sampling is disabled

<0-4095>

If parameter sample_rate is 'N' then 1/N of packets is sampled.

<14-200>

Configures the size of the header of the sampled frame to be copied to the Queue for further processing. The Max header size ranges from 14 to 200 bytes.

<0-3600>

Configures the polling interval for the counter sampling. The accepted value for Counter Polling Interval ranges from 0 to 3600 seconds. Default value is 0 seconds which means polling is disabled.

Example:

```
Switch(sflow)# sampler 2 aLL 400 199 3600
Switch(sflow)# show sampler
```

sFlow Ports	sFlow Instance	Sampler Type	Sampling Rate	Max Hdr Size	Counter Polling Interval
1	1	None	0	128	0
2	1	ALL	400	199	3600
3	1	None	0	128	0
4	1	None	0	128	0

40.3 *show*

Using this command you can show sFlow.

Syntax:

```
show collector | sampler
```

Parameter:**collector**

Shows sFlow collector

sampler

Shows sFlow sampler

Example:

```
Switch(sflow)# show collector
              Configured      Current
              -----
Collector Id  1                1
IP Type       IPv4             IPv4
IP Address    0.0.0.0          0.0.0.0
Port         6343             6343
Time Out      0                0 Timer is still alive!
Datagram Size 1400            1400

Switch(sflow)# show sampler
sFlow sFlow  Sampler  Sampling  Max Hdr  Counter  Polling
Ports Instance Type    Rate    Size    Interval
-----
  1      1      None     0      128         0
  2      1      None     0      128         0
  3      1      None     0      128         0
  4      1      None     0      128         0
```

41 Single IP Commands

Single IP Management (SIM), a simple and useful method to optimize network utilities and management, is designed to manage a group of switches as a single entity, called an SIM group. Implementing the SIM feature will have the following advantages for users

- Simplify management of small workgroups or wiring closets while scaling networks to handle increased bandwidth demand.
- Reduce the number of IP addresses needed on the network.
- Virtual stacking structure - Eliminate any specialized cables for stacking and remove the distance barriers that typically limit topology options when using other stacking technology.

Command	Function
connect	Connect to slave switch
connect	Configure single ip group name
mode	Configure single ip mode
show	Show single ip information

41.1 *connect*

Using this command you can connect to slave switch.

Syntax:

```
connect <1-16>
```

Parameter:

<1-16>

Slave switch index

Example:

```
Switch(sip)# connect 1
```

41.2 *group-name*

Using this command you can configure single IP group name.

Syntax:

```
group-name <name>
```

Parameter:**<name>**

Up to 64 characters describing group name

Example:

```
Switch(sip)# group-name david
Switch(sip)# show config
Mode           : Disabled
Group Name     : david
```

41.3 *mode*

Using this command you can configure single IP mode.

Syntax:

```
mode disable|master|slave
```

Parameter:**disable**

Disables single ip operation.

master

Configure as master.

slave

Configure as slave

Example:

```
Switch(sip)# mode master
Switch(sip)# show c
Mode           : Master
Group Name     : david
```

41.4 *show*

Using this command you can show single IP information.

Syntax:

```
show config|info
```

Parameter:**config**

Shows single ip configuration.

info

Shows single ip group information.

Example:

```
Switch(sip)# show config
Mode           : Disabled
Group Name     : VirtualStack

Switch(sip)# show info
Index  Model Name      MAC Address
-----

```

42 SMTP Commands

The function, is used to set a Alarm trap when the switch alarm then you could set the SMTP server to send you the alarm mail.

Command	Function
delete	Delete commands
level	Delete commands
mail-address	Configure email user name
return-path	Configure email sender
sender	Configure email sender
server	Configure email server
show	Show email configuration
username	Show DHCP snooping information

42.1 *delete*

Using this command you can delete command.

Syntax:

```
delete mail-address <1-6>  
delete return-path|sender|server|username
```

Parameter:**mail-address**

Delete email address.

<1-6>

Delete email address id.

return-path

Delete return path.

sender

Delete sender.

server

Delete email server.

username

Delete username and password.

Example:

```
Switch(smtp)# delete mail-address 2
Switch(smtp)# show
Mail Server      :
User Name       :
Password        :
Severity level   : Info
Sender          :
Return Path     :
Email Address 1  :
Email Address 2  :
Email Address 3  :
Email Address 4  :
Email Address 5  :
Email Address 6  :
```

42.2 level

Using this command you can configure Severity level.

Syntax:

```
level <0-7>
```

Parameter:**<0-7>**

Severity level

<0> Emergency: system is unusable

<1> Alert: action must be taken immediately

<2> Critical: critical conditions

<3> Error: error conditions

<4> Warning: warning conditions

<5> Notice: normal but significant condition

<6> Informational: informational messages

<7> Debug: debug-level messages

Example:

```
Switch(smtp)# level 7
Switch(smtp)# show
Mail Server      :
User Name       :
Password        :
Severity level   : Debug
Sender          :
Return Path     :
Email Address 1  :
Email Address 2  :
Email Address 3  :
```

```
Email Address 4 :  
Email Address 5 :  
Email Address 6 :
```

42.3 mail-address

Using this command you can configure email user name.

Syntax:

```
mail-address <1-6> <mail-address>
```

Parameter:

<1-6>

Email address index.

<mail-address>

Up to 47 characters describing mail address.

Example:

```
Switch(smtp)# mail-address 6 david@tech.com.tw  
Switch(smtp)# show  
Mail Server      :  
User Name       :  
Password        :  
Severity level   : Debug  
Sender          :  
Return Path     :  
Email Address 1  :  
Email Address 2  :  
Email Address 3  :  
Email Address 4  :  
Email Address 5  :  
Email Address 6  : david@tech.com.tw
```

42.4 return-path

Using this command you can configure the address of email sender.

Syntax:

```
return-path <return-path>
```

Parameter:

<return-path>

Up to 47 characters describing return path.

Example:

```
Switch(smtp)# return-path david@tech.com.tw
Switch(smtp)# show
Mail Server      :
User Name       :
Password        :
Severity level   : Debug
Sender          :
Return Path      : david@tech.com.tw
Email Address 1  :
Email Address 2  :
Email Address 3  :
Email Address 4  :
Email Address 5  :
Email Address 6  : david@tech.com.tw
```

42.5 sender

Using this command you can configure email sender.

Syntax:

```
sender <sender>
```

Parameter:**<sender>**

Up to 47 characters describing sender.

Example:

```
Switch(smtp)# sender tech
Switch(smtp)# show
Mail Server      :
User Name       :
Password        :
Severity level   : Debug
Sender          : david
Return Path      : david@mail. tech.com.tw
Email Address 1  :
Email Address 2  :
Email Address 3  :
Email Address 4  :
Email Address 5  :
Email Address 6  : david@tech.com.tw
```

42.6 server

Using this command you can configure email server.

Syntax:

```
server <server>
```

Parameter:

<server>

Up to 47 characters describing email server.

Example:

```
Switch(smtp)# server davidserver
Switch(smtp)# show
Mail Server      : davidserver
User Name       :
Password        :
Severity level   : Debug
Sender          : davidtech
Return Path      : david@mail.davidtech.com.tw
Email Address 1  :
Email Address 2  :
Email Address 3  :
Email Address 4  :
Email Address 5  :
Email Address 6  : jack@davidtech.com.tw
```

42.7 show

Using this command you can show email configuration.

Syntax:

```
show
```

Example:

```
Switch(smtp)# show
Mail Server      :
User Name       :
Password        :
Severity level   : Info
Sender          :
Return Path      :
Email Address 1  :
Email Address 2  :
Email Address 3  :
Email Address 4  :
Email Address 5  :
Email Address 6  :
```

42.8 username

Using this command you can configure email user name.

Syntax:

```
username <username> <password>
```

Parameter:**<username>**

Up to 47 characters describing user name.

<password>

Up to 47 characters describing password

Example:

```
Switch(smtp)# username david 1111
Switch(smtp)# show
Mail Server      : davidserver
User Name       : david
Password        : *****
Severity level   : Debug
Sender          : davidtech
Return Path     : david@mail.davidtech.com.tw
Email Address 1  :
Email Address 2  :
Email Address 3  :
Email Address 4  :
Email Address 5  :
Email Address 6  : rose@davidtech.com.tw
```

43 SNMP Commands

Any Network Management System (NMS) running the Simple Network Management Protocol (SNMP) can manage the Managed devices equipped with SNMP agent, provided that the Management Information Base (MIB) is installed correctly on the managed devices. The SNMP is a protocol that is used to govern the transfer of information between SNMP manager and agent and traverses the Object Identity (OID) of the management Information Base (MIB), described in the form of SMI syntax. SNMP agent is running on the switch to response the request issued by SNMP manager.

Basically, it is passive except issuing the trap information. The switch supports a switch to turn on or off the SNMP agent. If you set the field SNMP "Enable", SNMP agent will be started up. All supported MIB OIDs, including RMON MIB, can be accessed via SNMP manager. If the field SNMP is set "Disable", SNMP agent will be de-activated, the related Community Name, Trap Host IP Address, Trap and all MIB counters will be ignored.

Command	Function
access	Configure SNMP access
community	Configure SNMP community
delete	Delete command
engine-id	Set SNMP Engine ID
getcommunity	Configure SNMP Get Community
group	Configure SNMP groups
mode	Enable/Disable SNMP mode
setcommunity	Configure SNMP Set Community
show	Show SNMP command
trap	Configure SNMP trap
user	Configure SNMP users
view	Configure SNMP views

43.1 access

Using this command you can configure SNMP access.

Syntax:

```
access <groupname> <any|usmno> (AuthNoPriv|AuthPriv|NoAuthNoPriv) <read-view-name>  
<write-view-name>  
access <groupname> <v1|v2c> NoAuthNoPriv <read-view-name> <write-view-name>
```

Parameter:

<groupname>

max 32 chars.

any

Security Model.

usm

Security Model.

v1

Security Model.

v2c

Security Model.

AuthNoPriv

Security Level. If security_model is not usm, the security_level value must be NoAuthNoPriv.

AuthPriv

Security Level. If security_model is not usm, the security_level value must be NoAuthNoPriv.

NoAuthNoPriv

Security Level. If security_model is not usm, the security_level value must be NoAuthNoPriv.

<read-view-name>

The scope for a specified instance can read, None is reserved for Empty.

<write-view-name>

The scope for a specified instance can write, None is reserved for Empty.

Example:

```
Switch(snmpp)# access g usm noAuthNoPriv v v
Switch(snmpp)# show access
```

SNMPv3 Accesses Table:

Idx	Group Name	Model	SecurityLevel	Read View Name	Write View Name
1	g	usm	NoAuth, NoPriv	v	v

43.2 community

Using this command you can configure SNMP community.

Syntax:

```
community <community> <username> <ip-address> <ip-mask>
```

Parameter:**<community>**

max 32 chars.

<username>

max 32 chars.

<ip-address>

SNMP access source ip.

<ip-mask>

SNMP access source address mask.

Example:

```
Switch(snmp)# community david pm 192.168.6.127 255.255.255.0
Switch(snmp)# show community
```

SNMP Community Table:				
Idx	Community	UserName	Source IP	Source Mask
1	david	pm	192.168.6.127	255.255.255.0

Number of entries: 1

43.3 delete

Using this command you can delete command.

Syntax:

```
delete access <1-14>
delete community <1-4>
delete group <1-14>
delete trap <1-6>
delete user <1-10>
delete view <1-48>
```

Parameter:**access**

Delete snmpv3 access entry

<1-14>

table index.

community

Delete community entry.

<1-4>

table index.

group

Delete snmpv3 groups entry.

<1-14>

table index.

trap

Delete trap entry.

<1-6>

table index.

user

Delete snmpv3 users entry.

<1-10>

table index.

view

Delete snmpv3 views entry.

<1-48>

table index.

Example:

```
Switch(snmp)# delete access 14
```

43.4 *engine-id*

Using this command you can set SNMP Engine ID.

Syntax:

```
engine-id <HEX>
```

Parameter:

<HEX>

the format may not be all zeros or all 'ff',and is restricted to 5–32 octet string.

Example:

```
Switch(snmp)# engine-id ffffffffffffffffff
```

43.5 *getcommunity*

Using this command you can configure SNMP Get Community.

Syntax:

```
getcommunity <community>
```

Parameter:

<community>

max 32 chars, default : public.

Example:

```
Switch(snmplib)# getcommunity rose
Switch(snmplib)# show snmp

SNMP Configuration
-----
Get Community       : rose
Set Community Mode  : Enable
Set Community       : jack
```

43.6 group

Using this command you can configure SNMP groups.

Syntax:

```
group <username> usm|v1|v2c <groupname>
```

Parameter:**<username>**

max 32 chars

usm

Security Model.

v1

Security Model

v2c

Security Model

<groupname>

max 32 chars

Example:

```
Switch(snmplib)# group pm v1 ccc
Switch(snmplib)# show group

SNMPv3 Groups Table:
Idx Model Security Name          Group Name
-----
1   v1    pm                    ccc

Number of entries: 1

Switch(snmplib)# group pm v2c aaa
Switch(snmplib)# show group

SNMPv3 Groups Table:
Idx Model Security Name          Group Name
-----
1   v2c    pm                    aaa
```

43.7 *mode*

Using this command you can Enable/Disable SNMP mode.

Syntax:

```
mode disable|enable
```

Parameter:

disable

Disable SNMP mode.

enable

Enables SNMP mode.

Example:

```
Switch(snmp)# mode enable
Switch(snmp)# show mode

SNMPv3 State Show
SNMP State           : Enabled
SNMPv3 Engine ID     : 80001455030040c7232600
```

43.8 *setcommunity*

Using this command you can configure SNMP Set Community.

Syntax:

```
setcommunity disable
setcommunity enable <community>
```

Parameter:

disable

Disables SNMP Set Community.

enable

Enables SNMP Set Community.

<community>

max 32 chars, default : private

Example:

```
Switch(snmp)# setcommunity enable jack
Switch(snmp)# show snmp

SNMP Configuration
-----
```

```
Get Community      : eee
Set Community Mode : Enable
Set Community      : jack
```

43.9 *show*

Using this command you can show SNMP command.

Syntax:

```
show access|community|group|mode|snmp|trap|user|view
```

Parameter:

access

Shows snmpv3 access entry.

community

Shows snmpv3 community entry.

group

Shows snmpv3 groups entry

mode

Shows snmp configuration.

snmp

Shows snmp community configuration.

trap

Shows snmp trap entry.

user

Shows snmpv3 users entry.

view

Shows snmpv3 views entry.

Example:

```
Switch(snmp)# show access

SNMPv3 Accesses Table:
Idx   Group Name   Model SecurityLevel   Read View Name   Write View Name
-----
Number of entries: 0

Switch(snmp)# show community

SNMP Community Table:
Idx Community   UserName           Source IP           Source Mask
-----
1   david        pm                 192.168.6.127      255.255.255.0
Number of entries: 1
```

43.10 *trap*

Using this command you can configure SNMP trap.

Syntax:

```
trap <1-6> v2 ipv4|ipv6 <ip-address> <1-65535> <0-7> <community>
trap <1-6> v3 ipv4|ipv6 <ip-address> <1-65535> <0-7> <secname> NoAuthNoPriv
trap <1-6> v3 ipv4|ipv6 <ip-address> <1-65535> <0-7> <secname> AuthNoPriv MD5|SHA <auth-pwd>
trap <1-6> v3 ipv4|ipv6 <ip-address> <1-65535> <0-7> <secname> AuthPriv SHA|MD5 <auth-pwd>
DES <priv-pwd>
```

Parameter:

<1-6>

trap index: 1–6.

v2

version.

v3

version.

ipv4

Trap host IP type.

ipv6

Trap host IP type.

<ip-address>

Trap host IP address.

<1-65535>

trap port.

<0-7>

Severity level:

<0> Emergency: system is unusable

<1> Alert: action must be taken immediately

<2> Critical: critical conditions

<3> Error: error conditions

<4> Warning: warning conditions

<5> Notice: normal but significant condition

<6> Informational: informational messages

<7> Debug: debug-level messages

<community>

Community.

<secname>

Security name

NoAuthNoPriv

Security_Level

AuthNoPriv

Security_Level

AuthPriv

Security_Level

MD5

Authentication Protocol

SHA

Authentication Protocol

<auth-pwd>

SHA Authentication Password is restricted to 8–40

MD5 Authentication Password is restricted to 8–32

DES

Privacy Protocol

<priv-pwd>

MD5 Authentication Password is restricted to 8–32

Example:

```
Switch(snmp)# trap 2 v3 ipv4 192.168.2.110 2 5 asvsd AuthNoPriv SHA 123456789
Switch(snmp)# trap 2 v2 ipv4 192.168.6.127 65535 7 aaa
Switch(snmp)# show trap
SNMPv3 Trap Host Configuration:
```

	Priv.		Community	Severity	Auth.
No Ver	Server	IP	Port	Security	Name
Protocol				Level	Protocol

1					
2	v2c	192.168.6.127	65535	aaa	Debug
3					
4					
5					
6					

43.11 user

Using this command you can configure SNMP users.

Syntax:

```
user <username> NoAuthNoPriv
user <username> AuthNoPriv MD5|SHA <auth-pwd>
user <username> AuthPriv SHA|MD5 <auth-pwd> DES <priv-pwd>
```

Parameter:**<username>**

max 32 chars.

NoAuthNoPriv

Security_Level.

AuthNoPriv

Security_Level.

AuthPriv

Security_Level.

MD5

Authentication Protocol.

SHA

Authentication Protocol.

<auth-pwd>

SHA Authentication Password is restricted to 8–40

MD5 Authentication Password is restricted to 8–32

DES

Privacy Protocol

<priv-pwd>

MD5 Authentication Password is restricted to 8–32

Example:

```
Switch(snmp)# user wade authnoPriv md5 12345678
Switch(snmp)# show user

SNMPv3 Users Table:
Index User Name                Security Level Auth Priv
-----
1      wade                    AuthNoPriv    MD5   None

Number of entries: 1
```

43.12 view

Using this command you can configure SNMP views.

Syntax:

```
view <viewname> excluded|included <oid-subtree>
```

Parameter:**<viewname>**

max 32 chars.

excluded

view_type.

included

view_type.

<oid-subtree>

The OID defining the root of the subtree.

Example:

```
Switch(snm)# view viewdavid included .1.3.6.1.2
Switch(snm)# show view

SNMPv3 Views Table:
Idx View Name                               View Type OID Subtree
-----
1    viewdavid                               included .1.3.6.1.2
```

44 SSH Commands

This section shows you how to use SSH (Secure SHell) to securely access the Switch. SSH is a secure communication protocol that combines authentication and data encryption to provide secure encrypted communication.

Command	Function
mode	Configure the SSH mode
show	Show SSH configuration

44.1 *mode*

Using this command you can configure the SSH mode.

Syntax:

```
mode disable|enable
```

Parameter:

disable

Disables SSH mode operation.

enable

Enables SSH mode operation.

Example:

```
Switch(ssh)# mode enable
Switch(ssh)# show
SSH Mode : Enabled
```

44.2 *show*

Using this command you can show SSH configuration.

Syntax:

```
show
```

Example:

```
Switch(ssh)# show
SSH Mode : Enabled
```

45 STP Commands

The Spanning Tree Protocol (STP) can be used to detect and disable network loops, and to provide backup links between switches, bridges or routers. This allows the switch to interact with other bridging devices (that is, an STP-compliant switch, bridge or router) in your network to ensure that only one route exists between any two stations on the network, and provide backup links which automatically take over when a primary link goes down.

STP - STP uses a distributed algorithm to select a bridging device (STP- compliant switch, bridge or router) that serves as the root of the spanning tree network. It selects a root port on each bridging device (except for the root device) which incurs the lowest path cost when forwarding a packet from that device to the root device. Then it selects a designated bridging device from each LAN which incurs the lowest path cost when forwarding a packet from that LAN to the root device. All ports connected to designated bridging devices are assigned as designated ports. After determining the lowest cost spanning tree, it enables all root ports and designated ports, and disables all other ports. Network packets are therefore only forwarded between root ports and designated ports, eliminating any possible network loops.

Once a stable network topology has been established, all bridges listen for Hello BPDUs (Bridge Protocol Data Units) transmitted from the Root Bridge. If a bridge does not get a Hello BPDU after a predefined interval (Maximum Age), the bridge assumes that the link to the Root Bridge is down. This bridge will then initiate negotiations with other bridges to reconfigure the network to reestablish a valid network topology.

Command	Function
CName	Set MSTP Configuration name
FwdDelay	Set FwdDelay
MaxAge	Set Maxage
MaxHops	Set MaxHops
Statistics	Clear STP port statistics
Txhold	Set TxHold
Version	Set force-version
bpduFilter	Set edge port BPDU Filtering
bpduGuard	Set edge port BPDU Guard
migrate-check	Set the STP mCheck (Migration Check) variable for ports
msti-vlan	Map Vlan ID(s) to an MSTI
p-AutoEdge	Set the STP autoEdge port parameter
p-bpduGuard	Set the bpduGuard port parameter
p-cost	Set the STP port instance path cost
p-edge	Set the STP adminEdge port parameter
p-mode	Set the STP enabling for a port
p-p2p	Set the STP point2point port parameter
p-priority	Set the STP port instance priority
priority	Set the bridge instance priority
r-role	Set the MSTP restrictedRole port parameter
r-tcn	Set the MSTP restrictedTcn port parameter
recovery	Set edge port error recovery timeout
show	Show Region config, MSTI vlan mapping, instance parameters and port parameters

45.1 *cname*

Using this command you can Set MSTP Configuration name.

Syntax:

```
cname <name> [<0-65535>]
```

Parameter:

<name>

A text string up to 32 characters long.

<0-65535>

MSTP revision-level (0~65535).

Example:

```
Switch(stp)# cName david 65535
Switch(stp)# show cName
Configuration name: david
Configuration rev.: 65535
```

45.2 *fwdDelay*

Using this command you can Set FwdDelay.

Syntax:

```
fwdDelay <4-30>
```

Parameter:

<4-30>

MSTP forward delay (4-30, and $\text{max_age} \leq (\text{forward_delay} - 1) * 2$).

Example:

```
Switch(stp)# fwdDelay 30
Switch(stp)# show instance
STP Configuration
Protocol Version: MSTP
Max Age          : 20
Forward Delay    : 30
Tx Hold Count    : 6
Max Hop Count    : 20
BPDU Filtering   : Disabled
BPDU Guard       : Disabled
Error Recovery   : 0 seconds
Error Recovery   : Disabled
```

45.3 *maxage*

Using this command you can Set Maxage.

Syntax:

```
maxage <6-40>
```

Parameter:

<6-40>

STP maximum age time (6-40, and max_age <= (forward_delay-1)*2).

Example:

```
Tx Hold Count      : 6
Max Hop Count      : 20
BPDU Filtering     : Disabled
BPDU Guard         : Disabled
Error Recovery     : 0 seconds
Error Recovery     : Disabled
```

45.4 *maxhops*

Using this command you can Set MaxHops.

Syntax:

```
maxhops <6-40>
```

Parameter:

<6-40>

STP BPDU MaxHops (6-40).

Example:

```
Switch(stp)# maxhops 38
Switch(stp)# show instance
STP Configuration
Protocol Version: MSTP
Max Age         : 39
Forward Delay   : 30
Tx Hold Count   : 6
Max Hop Count   : 38
BPDU Filtering  : Disabled
BPDU Guard      : Disabled
Error Recovery  : 0 seconds
Error Recovery  : Disabled
```

45.5 *statistics*

Using this command you can Clear STP port statistics.

Syntax:

```
statistics clear
```

Parameter:

clear

Clear the selected port statistics.

Example:

```
Switch(stp)# statistics clear
Port          Rx MSTP   Tx MSTP   Rx RSTP   Tx RSTP   Rx STP   Tx STP   Rx TCN
T
x TCN   Rx Ill.  Rx Unk.
-----
-
-----
```

45.6 *txhold*

Using this command you can Set TxHold.

Syntax:

```
txhold <1-10>
```

Parameter:

<1-10>

STP Transmit Hold Count (1-10).

Example:

```
Switch(stp)# txhold 9
Switch(stp)# show instance
STP Configuration
Protocol Version: MSTP
Max Age          : 39
Forward Delay    : 30
Tx Hold Count    : 9
Max Hop Count    : 38
BPDU Filtering   : Disabled
BPDU Guard       : Disabled
Error Recovery   : 0 seconds
Error Recovery    : Disabled
```

45.7 *version*

Using this command you can Set force-version.

Syntax:

```
version mstp|rstp|stp
```

Parameter:**mstp**

Multiple Spanning Tree Protocol

rstp

Rapid Spanning Tree Protocol

stp

Spanning Tree Protocol

Example:

```
Switch(stp)# version stp
Switch(stp)# show instance
STP Configuration
Protocol Version: Compatible (STP)
Max Age          : 39
Forward Delay    : 30
Tx Hold Count    : 9
Max Hop Count    : 38
BPDU Filtering   : Disabled
BPDU Guard       : Disabled
Error Recovery   : 0 seconds
Error Recovery   : Disabled
```

45.8 *bpdufilter*

Using this command you can Set edge port BPDU Filtering what you set on the switch.

Syntax:

```
bpdufilter disable|enable
```

Parameter:**disable**

Disables BPDU Filtering for Edge ports.

enable

Enables BPDU Filtering for Edge ports.

Example:

```
Switch(stp)# bpdufilter enable
Switch(stp)# show instance
STP Configuration
Protocol Version: Compatible (STP)
Max Age       : 39
Forward Delay : 30
Tx Hold Count : 9
Max Hop Count : 38
BPDU Filtering : Enabled
BPDU Guard    : Disabled
Error Recovery : 0 seconds
Error Recovery : Disabled
```

45.9 *bpduguard*

Using this command you can Set edge port BPDU Guard.

Syntax:

```
bpduguard disable|enable
```

Parameter:**disable**

Disables BPDU Guard for Edge ports.

enable

Enables BPDU Guard for Edge ports.

Example:

```
Switch(stp)# bpduguard enable
Switch(stp)# show instance
STP Configuration
Protocol Version: Compatible (STP)
Max Age       : 39
Forward Delay : 30
Tx Hold Count : 9
Max Hop Count : 38
BPDU Filtering : Enabled
BPDU Guard    : Enabled
Error Recovery : 0 seconds
Error Recovery : Disabled
```

45.10 *migrate-check*

Using this command you can Set the STP mCheck (Migration Check) variable for ports.

Syntax:

```
migrate-check <port-list>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(stp)# migrate-check 1
```

45.11 *msti-vlan*

Using this command you can Map Vlan ID(s) to an MSTI.

Syntax:

```
msti-vlan add <0-7> <1-4094>  
msti-vlan del <0-7>
```

Parameter:**add**

add a VLAN to a MSTI.

del

clear MSTP MSTI VLAN mapping configuration.

<0-7>

STP bridge instance no (0-7, CIST=0, MSTI1=1, ...)

<1-4094>

available from 1 to 4094

Example:

```
Switch(stp)# msti-vlan add 1 4094  
Switch(stp)# show msti-vlan  
MSTI  VLANs mapped to MSTI  
----  -  
MSTI1  4094  
MSTI2  No VLANs mapped  
MSTI3  No VLANs mapped  
MSTI4  No VLANs mapped  
MSTI5  No VLANs mapped  
MSTI6  No VLANs mapped  
MSTI7  No VLANs mapped
```

45.12 *p-autoEdge*

Using this command you can Set the STP autoEdge port parameter.

Syntax:

```
p-autoEdge aggregations|<port-list> disable|enable
```

Parameter:

aggregations

available value is for aggregated port.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables MSTP autoEdges

enable

Enables MSTP autoEdge

Example:

```
Switch(stp)# p-autoEdge aggregations enable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Enabled

```
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
1	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto
2	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto
3	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto

```
Switch(stp)# p-autoEdge 1 disable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Enabled

```
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
1	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled	Auto
2	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto
3	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto

45.13 *p-bpduguard*

Using this command you can Set the bpduGuard port parameter.

Syntax:

```
p-bpduguard aggregations|<port-list> disable|enable
```

Parameter:

aggregations

available value is for aggregated port.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables BPDU Guard

enable

Enables BPDU Guard

Example:

```
Switch(stp)# p-bpduGuard aggregations enable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Disabled	Disabled	Disabled	Disabled	Disabled	Enabled	Enabled
Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
1	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled	Auto
2	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled	Auto
3	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled	Auto

```
Switch(stp)# p-bpduGuard 1 enable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Disabled	Disabled	Enabled	Disabled	Disabled	Enabled	Enabled
Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
1	Disabled	Disabled	Disabled	Disabled	Disabled	Enabled	Auto
2	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto
3	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto

45.14 *p-cost*

Using this command you can Set the STP port instance path cost.

Syntax:

```
p-cost <0-7> aggregations|<port-list> <0-200000000>
```

Parameter:**<0-7>**

STP bridge instance no (0-7, CIST=0, MSTI1=1, ...)

aggregations

available value is for aggregated port.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-200000000>

STP port path cost (1-200000000) or The value zero means auto status

Example:

```
Switch(stp)# p-cost 0 aggregations 2000000
```

```
Switch(stp)# show p-config 0
```

MSTI	Port	Path Cost	Priority
CIST	Aggr	2000000	128

MSTI	Port	Path Cost	Priority
CIST	1	Auto	128
CIST	2	Auto	128
CIST	3	Auto	128

```
Switch(stp)# p-cost 1 3 9999
```

```
Switch(stp)# show p-config 1
```

MSTI	Port	Path Cost	Priority
MSTI1	Aggr	Auto	128

MSTI	Port	Path Cost	Priority
MSTI1	1	Auto	128
MSTI1	2	Auto	128
MSTI1	3	9999	128

45.15 p-edge

Using this command you can Set the STP adminEdge port parameter.

Syntax:

```
p-edge aggregations|<port-list> disable|enable
```

Parameter:**aggregations**

available value is for aggregated port.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables MSTP protocol

enable

Enables MSTP protocol

Example:

```
Switch(stp)# p-edge aggregations enable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Disabled	Enabled	Enabled	Disabled	Disabled	Enabled	Enabled

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
1	Disabled	Disabled	Disabled	Disabled	Disabled	Enabled	Auto
2	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto

45.16 *p-mode*

Using this command you can Set the STP enabling for a port.

Syntax:

```
p-mode aggregations|<port-list> disable|enable
```

Parameter:**aggregations**

available value is for aggregated port.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables MSTP protocol

enable

Enables MSTP protocol

Example:

```
Switch(stp)# p-mode aggregations enable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Enabled	Disabled	Enabled	Disabled	Disabled	Enabled	Enabled

```
Port Mode AdminEdge AutoEdge restrRole restrTcn bpduGuard Point2point
-----
1 Disabled Disabled Disabled Disabled Disabled Enabled Auto
2 Disabled Disabled Enabled Disabled Disabled Disabled Auto
```

45.17 p-p2p

Using this command you can Set the STP point2point port.

Syntax:

```
p-p2p aggregations|<port-list> auto|disable|enable
```

Parameter:**aggregations**

available value is for aggregated port.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

auto

Automatic MSTP point2point detection

disable

Disables MSTP point2point

enable

Enables MSTP point2point

Example:

```
Switch(stp)# p-p2p aggregations auto
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Enabled	Disabled	Enabled	Disabled	Disabled	Enabled	Auto

```
Port Mode AdminEdge AutoEdge restrRole restrTcn bpduGuard Point2point
-----
1 Disabled Disabled Disabled Disabled Disabled Enabled Auto
2 Disabled Disabled Enabled Disabled Disabled Disabled Auto
```

```
Switch(stp)# p-p2p 2 disable
```

```
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Enabled	Disabled	Enabled	Disabled	Disabled	Enabled	Auto
1	Disabled	Disabled	Disabled	Disabled	Disabled	Enabled	Auto
2	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Disabled
3	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto

45.18 *p-priority*

Using this command you can Set the STP port instance priority.

Syntax:

```
p-priority <0-7> aggregations|<port-list> <0-240>
```

Parameter:

<0-7>

STP bridge instance no (0-7, CIST=0, MSTI1=1, ...)

aggregations

available value is for aggregated port.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-240>

STP bridge priority (0/16/32/48/.../224/240)

Example:

```
Switch(stp)# p-priority 3 aggregations 240
Switch(stp)# show p-config 3
```

MSTI	Port	Path Cost	Priority
MSTI3	Aggr	Auto	240

MSTI	Port	Path Cost	Priority
MSTI3	1	Auto	128
MSTI3	2	Auto	128

```
Switch(stp)# p-priority 1 2 224
Switch(stp)# show p-config 1
```

MSTI	Port	Path Cost	Priority
MSTI1	Aggr	Auto	128

MSTI	Port	Path Cost	Priority
----	----	-----	-----
MSTI1	1	Auto	128
MSTI1	2	Auto	224

45.19 *priority*

Using this command you can Set the bridge instance priority.

Syntax:

```
priority <0-7> <0-61440>
```

Parameter:

<0-7>

STP bridge instance no (0-7, CIST=0, MSTI1=1, ...)

<0-61440>

STP bridge priority (0/4096/8192/12288/.../57344/61440)

Example:

```
Switch(stp)# priority 0 61440
Switch(stp)# show priority
MSTI#   Bridge Priority
-----
CIST    61440
```

45.20 *r-role*

Using this command you can Set the MSTP restrictedRole port parameter.

Syntax:

```
r-role aggregations|<port-list> disable|enable
```

Parameter:

aggregations

available value is for aggregated port.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables MSTP restricted role

enable

Enables MSTP restricted role

Example:

```
Switch(stp)# r-role aggregations enable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Enabled	Disabled	Enabled	Enabled	Disabled	Enabled	Auto

```
Switch(stp)# r-role 2 enable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Enabled	Disabled	Enabled	Enabled	Disabled	Enabled	Auto
1	Disabled	Disabled	Disabled	Disabled	Disabled	Enabled	Auto
2	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Disabled
3	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Auto

```
Switch(stp)# r-role 2 enable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Enabled	Disabled	Enabled	Enabled	Disabled	Enabled	Auto
1	Disabled	Disabled	Disabled	Disabled	Disabled	Enabled	Auto
2	Disabled	Disabled	Enabled	Enabled	Disabled	Disabled	Disabled

45.21 *r-tcn*

Using this command you can Set the MSTP restrictedTcn port parameter.

Syntax:

```
r-tcn aggregations|<port-list> disable|enable
```

Parameter:**aggregations**

available value is for aggregated port.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables MSTP restricted TCN

enable

Enables MSTP restricted TCN

Example:

```
Switch(stp)# r-tcn aggregations enable
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
-----	-----	-----	-----	-----	-----	-----	-----
Aggr	Enabled	Disabled	Enabled	Enabled	Enabled	Enabled	Auto


```
Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
-----	-----	-----	-----	-----	-----	-----	-----
1	Disabled	Disabled	Disabled	Disabled	Disabled	Enabled	Auto
2	Disabled	Disabled	Enabled	Enabled	Disabled	Disabled	Disabled

45.22 recovery

Using this command you can Set edge port error recovery timeout.

Syntax:

```
recovery <30-86400>
```

Parameter:

<30-86400>

Time before error-disabled ports are reenabled (30-86400 seconds, 0 disables)

Example:

```
Switch(stp)# recovery 86400
Switch(stp)# show instance
STP Configuration
Protocol Version: Compatible (STP)
Max Age          : 39
Forward Delay    : 30
Tx Hold Count    : 9
Max Hop Count    : 38
BPDU Filtering   : Enabled
BPDU Guard       : Enabled
Error Recovery    : 86400 seconds
Error Recovery    : Disabled
```

45.23 show

Using this command you can Show Region config, MSTI vlan mapping, instance parameters and port parameters.

Syntax:

```
show CName|statistics|instance|msti-vlan|pconf|priority
show status|p-config <0-7>
```

Parameter:**CName**

Shows MSTP Configuration name

statistics

Shows STP port statistics

instance

Shows instance status

msti-vlan

Shows MSTP MSTI VLAN mapping configuration

pconf

Shows STP Port configuration

priority

Shows the bridge instance priority

status

Shows STP Bridge status

p-config

Shows the STP port instance configuration

<0-7>

STP bridge instance no (0-7, CIST=0, MSTI1=1, ...)

Example:

```
Switch(stp)# show cName
Configuration name: 00-40-c7-23-26-00
Configuration rev.: 0

Switch(stp)# show instance
STP Configuration
Protocol Version: MSTP
Max Age       : 20
Forward Delay : 15
Tx Hold Count : 6
Max Hop Count : 20
BPDU Filtering : Disabled
BPDU Guard    : Disabled
Error Recovery : 0 seconds
Error Recovery : Disabled

Switch(stp)# show pconf
```

Port	Mode	AdminEdge	AutoEdge	restrRole	restrTcn	bpduGuard	Point2point
Aggr	Disabled	Disabled	Enabled	Disabled	Disabled	Disabled	Enabled

```
Port  Mode      AdminEdge AutoEdge  restrRole restrTcn  bpduGuard Point2point
-----
1     Disabled  Disabled  Enabled   Disabled  Disabled  Disabled  Auto
2     Disabled  Disabled  Enabled   Disabled  Disabled  Disabled  Auto
3     Disabled  Disabled  Enabled   Disabled  Disabled  Disabled  Auto
```

46 Syslog Commands

The Syslog is a standard for logging program messages . It allows separation of the software that generates messages from the system that stores them and the software that reports and analyzes them. It can be used as well a generalized informational, analysis and debugging messages. It is supported by a wide variety of devices and receivers across multiple platforms.

Command	Function
clear	Clear syslog entry
FwdDelay	Configure syslog level
mode	Configure syslog mode
server	Configure syslog server IP address
show	Show syslog information

46.1 *clear*

clear.

Syntax:

```
clear
```

Example:

```
Switch(syslog)# clear
Switch(syslog)# show log
<0> Emergency: 0
<1> Alert      : 0
<2> Critical   : 0
<3> Error      : 0
<4> Warning    : 0
<5> Notice     : 0
<6> Info       : 0
<7> Debug      : 0
    All        : 0

ID   Level   Time           Message
-----
<none>
```

46.2 *level*

Using this command you can Configure syslog level.

Syntax:

```
level <0-7>
```

Parameter:**<0-7>**

Severity level.

<0> Emergency: system is unusable

<1> Alert: action must be taken immediately

<2> Critical: critical conditions

<3> Error: error conditions

<4> Warning: warning conditions

<5> Notice: normal but significant condition

<6> Informational: informational messages

<7> Debug: debug-level messages

Example:

```
Switch(syslog)# level 7
Switch(syslog)# show config
Server Mode      : Disabled
Server Address 1 :
Server Address 2 :
Syslog Level     : Debug
```

46.3 mode

Using this command you can Configure syslog mode.

Syntax:

```
mode disable|enable
```

Parameter:**disable**

Disable syslog mode.

enable

Enables syslog mode

Example:

```
Switch(syslog)# mode enable
Switch(syslog)# show config
Server Mode      : Enabled
Server Address 1 :
Server Address 2 :
Syslog Level     : Debug
```

46.4 *server*

Using this command you can Configure syslog server IP address.

Syntax:

```
server <1-2> <ip-hostname>
```

Parameter:

<1-2>

Syslog Server No.

<ip-hostname>

Syslog server IP address or host name

Example:

```
Switch(syslog)# server 2 192.168.6.1
Switch(syslog)# show config
Server Mode      : Enabled
Server Address 1 :
Server Address 2 : 192.168.6.1
Syslog Level     : Debug
```

46.5 *show*

Using this command you can Show syslog information.

Syntax:

```
show config
show detail-log <log-id>
show log [<0-7>]
```

Parameter:

config

Shows syslog configuration.

detail-log

Shows detailed syslog information

<log-id>

Log ID

log

Shows syslog entry

<0-7>

Shows syslog entry that match the level

Example:

```
Switch(syslog)# show config
Server Mode      : Disabled
Server Address 1 :
Server Address 2 :
Syslog Level     : Info

Switch(syslog)# show detail-log 2
ID      : 2
Level   : Warning
Time    : 2014-01-01 01:00:27
Message:

Link up on port 2

Switch(syslog)# show log 2
<0> Emergency: 0
<1> Alert    : 0
<2> Critical : 0
<3> Error    : 0
<4> Warning  : 8
<5> Notice   : 0
<6> Info     : 12
<7> Debug    : 0
All         : 20

ID   Level   Time                Message
-----
<none>
```

47 System Commands

After you login, the switch shows you the system information. This section is default and tells you the basic information of the system, including "Model Name", "System Description", "Contact", "Device Name", "System Up Time", "BIOS Version", "Firmware Version", "Hardware-Mechanical Version", "Serial Number", "Host IP Address", "Host Mac Address", "Device Port", "RAM Size", "Flash Size" and. With this information, you will know the software version used, MAC address, serial number, how many ports good and so on. This is helpful while malfunctioning.

Command	Function
contact	Configure system contact
location	Configure system location
name	Configure device name
show	Show system information

47.1 *contact*

Using this command you can Configure system contact.

Syntax:

```
contact <LINE>
```

Parameter:

<LINE>

Up to 255 characters describing system contact information.

Example:

```
GS-2326P(system)# contact LANCOM Systems GmbH +49 (0)2405 49936-0
GS-2326P(system)#
```

47.2 *location*

Using this command you can Configure system location.

Syntax:

```
location <LINE>
```

Parameter:

<LINE>

Up to 255 characters describing system location.

Example:

```
GS-2326P(system)# location Wuerselen, Germany
GS-2326P(system)#
```

47.3 *name*

Using this command you can Configure device name .

Syntax:

```
name <devicename>
```

Parameter:

<devicename>

Up to 255 characters describing device name.

Example:

```
Switch(system)# name MyDevice
Switch(system)#
```

47.4 *show*

Using this command you can Show system information .

Syntax:

```
show
```

Example:

```
MyDevice(system)# show
Model Name           : LANCOM GS-2326P
System Description   : 26-Port Layer-2 PoE+ Managed Gigabit Ethernet
Switch with 2x SFP
Location             : Wuerselen, Germany
Contact              : LANCOM Systems GmbH +49 (0)2405 49936-0
Device Name          : MyDevice
System Uptime        : 23:31:31
Current Time         : 2014-01-01 23:31:31
BIOS Version         : v1.00
Firmware Version     : v2.50
Hardware-Mechanical Version : v1.01-v1.01
Serial Number        : 275032000216
Host IP Address       : 192.168.2.110
Subnet Mask          : 255.255.255.0
Gateway IP Address   : 192.168.2.100
Host MAC Address     : 00-a0-57-1c-71-01
Console Baudrate     : 115200
```

RAM Size	: 64
Flash Size	: 16
CPU Load (100ms, 1s, 10s)	: 25%, 24%, 24%
Bridge FDB Size	: 8192 MAC addresses
Transmit Queue	: 8 queues per port
Maximum Frame Size	: 9600

48 Thermal Commands

The section describes the user to inspect and configure the current setting for controlling thermal protection. Thermal protection is used to protect the chip from getting overheated.

When the temperature exceeds the configured thermal protection temperature, ports will be turned off in order to decrease the power consumption. It is possible to arrange the ports with different priorities. Each priority can be given a temperature at which the corresponding ports shall be turned off.

Command	Function
port-priority	Configure the port priority
priority-temp	Configure the temperature at which the ports shall be shut down
show	Show thermal protection information

48.1 *port-priority*

Using this command you can Configure the port priority.

Syntax:

```
port-priority <port-list> <0-3>
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<0-3>

<0-3>:

Example:

```
Switch(thermal)# port-priority 1 3
Switch(thermal)# show
Priority  Temperature
-----
0         255 C
1         255 C
2         255 C
3         255 C

Port Priority Chip Temperature Port status
-----
1      3           60 C Port link operating normally
2      0           59 C Port link operating normally
3      0           59 C Port link operating normally
```

48.2 *priority-temp*

Using this command you can Configure the temperature at which the ports shall be shut down.

Syntax:

```
priority-temp <0-3> <0-255>
```

Parameter:

<0-3>

Port priority

<0-255>

The temperature at which the ports with the corresponding priority will be turned off

Example:

```
Switch(thermal)# priority-temp 1 99
Switch(thermal)# show
Priority  Temperature
-----
0        255 C
1        99 C
2        255 C
3        255 C

Port Priority Chip Temperature Port status
-----
1      3          59 C Port link operating normally
2      0          59 C Port link operating normally
3      0          59 C Port link operating normally
```

48.3 *show*

Using this command you can Show thermal protection information.

Syntax:

```
show
```

Example:

```
Switch(thermal)# show
Priority  Temperature
-----
0        255 C
1        255 C
2        255 C
3        255 C

Port Priority Chip Temperature Port status
-----
```

1	0	59 C Port link operating normally
2	0	59 C Port link operating normally
3	0	59 C Port link operating normally
4	0	59 C Port link operating normally

49 Time Commands

This section allows configuration of the switch time. Time configure is including Time Configuration and NTP Configuration

The switch provides manual and automatic ways to set the system time via NTP. Manual setting is simple and you just input "Year", "Month", "Day", "Hour", "Minute" and "Second" within the valid value range indicated in each item.

Command	Function
clock-source	Enable/Diable applicant administrative control
daylight	Set the GARP join timer configuration
delete	Set the GARP leave all timer configuration
manual	Set the GARP leave timer configuration
ntp	Configure NTP server
show	Show the GARP configuration
time-zone	Configure system time zone

49.1 *clock-source*

Using this command you can configure the clock source.

Syntax:

```
clock-source local|ntp
```

Parameter:

local

Local settings.

ntp

Use NTP to synchronize system clock.

Example:

```
Switch(time)# clock-source ntp
Switch(time)# show daylight
Clock Source       : NTP Server
Local Time         : 2014-01-01 07:19:44 (YYYY-MM-DD HH:MM:SS)
Time Zone Offset   : 0 (min)
Daylight Savings   : Disabled
```

49.2 *daylight*

Using this command you can indicates the Daylight Savings operation.

Syntax:

```
daylight disable
daylight enable <1-1440> By-dates <YYYY:MM:DD> <HH:MM> <YYYY:MM:DD> <HH:MM>
daylight enable <1-1440> Recurring <DAY> <WORD> <MONTH> <HH:MM> <DAY> <WORD> <MONTH> <HH:MM>
```

Parameter:**disable**

Disables Daylight Savings operation.

enable

Enables Daylight Savings operation.

<1-1440>

Minute. Time Set Offset.

By-dates

Manually enter day and time that DST starts and ends

<YYYY:MM:DD>

Day that DST starts

<HH:MM>

Time that DST starts

<YYYY:MM:DD>

Day that DST ends

<HH:MM>

Time that DST ends

Recurring

DST occurs on the same date every year

<DAY>

Sun, Mon, Tue, Wed, Thu, Fri, Sat
at which DST begins every year

<WORD>

first, 2, 3, 4, last
at which DST begins every year

<MONTH>

Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec

<HH:MM>

The time at which DST begins every year

<DAY>

Sun, Mon, Tue, Wed, Thu, Fri, Sat
at which DST ends every year

<WORD>

first, 2, 3, 4, last
at which DST ends every year

<MONTH>

Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec

at which DST ends every year

<HH:MM>

The time at which DST ends every year

Example:

```
Switch(time)# daylight enable 1440 by-dates 2015:03:01 10:00 2015:04:01
09:00
Switch(time)# show daylight
Clock Source      : NTP Server
Local Time        : 2014-01-01 07:23:21 (YYYY-MM-DD HH:MM:SS)
Time Zone Offset  : 0 (min)
Daylight Savings  : Enabled
Time Set Offset   : 1440 (min)
Daylight Savings Type : By dates
From              : 2015-03-01 10:00 (YYYY-MM-DD HH:MM)
To               : 2015-04-01 09:00 (YYYY-MM-DD HH:MM)

Switch(time)# daylight enable 1000 recurring wed 2 jan 11:00 sun 3 may 12:00
Switch(time)# show daylight
Clock Source      : NTP Server
Local Time        : 2014-01-01 07:28:43 (YYYY-MM-DD HH:MM:SS)
Time Zone Offset  : 0 (min)
Daylight Savings  : Enabled
Time Set Offset   : 1000 (min)
Daylight Savings Type : Recurring
From              : Day:Wed Week:2      Month:Jan Time:11:00
To               : Day:Sun Week:3      Month:May Time:12:00
```

49.3 delete

Using this command you can delete NTP server.

Syntax:

```
delete <1-5>
```

Parameter:

<1-5>

NTP server index.

Example:

```
Switch(time)# delete 1
```

49.4 *manual*

Using this command you can configure system time manually.

Syntax:

```
manual <YYYY:MM:DD> <HH:MM:SS>
```

Parameter:

<YYYY:MM:DD>

Date of system, example: 2014:01:12.

<HH:MM:SS>:

Time, example: 23:10:55.

Example:

```
Switch(time)# manual 2014:01:12 10:00:00
Switch(time)# show daylight
Clock Source      : Local Settings
Local Time        : 2014-01-12 10:00:07 (YYYY-MM-DD HH:MM:SS)
Time Zone Offset  : 0 (min)
Daylight Savings  : Enabled
Time Set Offset   : 1000 (min)
Daylight Savings Type : Recurring
From              : Day:Wed Week:2      Month:Jan Time:11:00
To                : Day:Sun Week:3      Month:May Time:12:00
```

49.5 *ntp*

Using this command you can configure NTP server.

Syntax:

```
ntp <1-5> <ipv6-address>|<ip-hostname>
```

Parameter:

<1-5>

NTP server index.

<ipv6-address>

NTP server IPv6 address.

IPv6 address is in 128-bit records represented as eight fields of up to four hexadecimal digits with a colon separate each field (":"). For example, **fe80::215:c5ff:fe03:4dc7**. The symbol "::" is a special syntax that can be used as a shorthand way of representing multiple 16-bit groups of contiguous zeros; but it can only appear once. It also used a following legally IPv4 address. For example, **::192.1.2.34**.

<ip-hostname>

NTP server IP address or hostname

Example:

```
Switch(time)# ntp 1 64.90.182.55
Switch(time)# show ntp
Index   Server IP host address or a host name string
-----
1       64.90.182.55
```

49.6 *show*

Using this command you can show time information.

Syntax:

```
show daylight|ntp
```

Parameter:**daylight**

Shows time information.

ntp

Shows NTP information.

Example:

```
Switch(time)# show daylight
Clock Source      : Local Settings
Local Time        : 2014-01-01 07:17:29 (YYYY-MM-DD HH:MM:SS)
Time Zone Offset  : 0 (min)
Daylight Savings  : Disabled

Switch(time)# show ntp
Index   Server IP host address or a host name string
-----
1
2
3
4
5
```

49.7 *time-zone*

Using this command you can configure system time zone.

Syntax:

```
time-zone -<HH:MM>| [+]<HH:MM>
```

Parameter:**<HH:MM>**

The time difference between GMT and local time, the possible value is from GMT-12:00 to GMT+12:00.

Example:

```
Switch(time)# time-zone 01:00
Switch(time)# show daylight
Clock Source      : NTP Server
Local Time        : 2014-01-12 11:14:24 (YYYY-MM-DD HH:MM:SS)
Time Zone Offset  : 60 (min)
Daylight Savings  : Enabled
Time Set Offset   : 1000 (min)
Daylight Savings Type : Recurring
From              : Day:Wed Week:2      Month:Jan Time:11:00
To                : Day:Sun Week:3      Month:May Time:12:00
```

50 UPnP Commands

UPnP is an acronym for Universal Plug and Play. The goals of UPnP are to allow devices to connect seamlessly and to simplify the implementation of networks in the home (data sharing, communications, and entertainment) and in corporate environments for simplified installation of computer components.

Command	Function
duration	Configure the advertising duration
mode	Configure UPnP mode
show	Show UPnP configuration
ttl	Configure the TTL value of the IP header in SSDP message

50.1 *duration*

Using this command you can Configure the advertising duration.

Syntax:

```
duration <100-86400>
```

Parameter:

<100-86400>

UPnP duration range.

Example:

```
Switch(upnp)# duration 86400
Switch(upnp)# show
UPnP Mode           : Disabled
UPnP TTL             : 4
UPnP Advertising Duration : 86400
```

50.2 *mode*

Using this command you can Configure UPnP mode.

Syntax:

```
mode disable|enable
```

Parameter:

disable

Disables UPnP.

enable

Enables UPnP

Example:

```
Switch(upnp)# mode enable
Switch(upnp)# show
UPnP Mode           : Enabled
UPnP TTL            : 4
UPnP Advertising Duration : 86400
```

50.3 *show*

Using this command you can Show UPnP configuration.

Syntax:*show***Example:**

```
Switch(upnp)# show
UPnP Mode           : Enabled
UPnP TTL            : 4
UPnP Advertising Duration : 86400
```

50.4 *ttl*

Using this command you can Configure the TTL value of the IP header in SSDP message.

Syntax:*t t l* <1-255>**Parameter:**

<1-255>

UPnP TTL value..

Example:

```
Switch(upnp)# ttl 255
Switch(upnp)# show
UPnP Mode           : Enabled
UPnP TTL            : 255
UPnP Advertising Duration : 86400
```

51 VCL Commands

VLAN Control List indicates two types of VLAN, which are MAC address-based VLAN and Protocol -based VLAN.

MAC address-based VLAN decides the VLAN for forwarding an untagged frame based on the source MAC address of the frame.

MAC-based VLANs group VLAN members by MAC address. With MAC-based VLAN configured, the device adds a VLAN tag to an untagged frame according to its source MAC address. MAC-based VLANs are mostly used in conjunction with security technologies such as 802.1X to provide secure, flexible network access for terminal devices.

Protocol -based VLAN supports Protocol including Ethernet LLC and SNAP Protocol.

Command	Function
delete	Delete command
mac-vlan	Configure MAC-based VLAN membership
protocol-vlan	Configure protocol-based VLAN
show	Show VCL status command

51.1 delete

Using this command you can Delete command.



You need to set MAC VLAN or Protocol VLAN first, then you could delete and clear the configuration.

Syntax:

```
delete mac-vlan <mac-address>
delete protocol-vlan protocol ethernet <0x0600-0xffff>
delete protocol-vlan protocol llc <0x00-0xff> <0x00-0xff>
delete protocol-vlan protocol snap <oui-address> <0x0000-0xffff>
delete protocol-vlan vlan <WORD>
```

Parameter:

mac-vlan

Delete MAC-based VLAN entry

<mac-address>

MAC address, format 0a-1b-2c-3d-4e-5f

protocol-vlan

Delete protocol-based VLAN entry

protocol

Delete protocol-based VLAN ethertype protocol to group mapping

Ethernet

Delete protocol-based VLAN Ethernet-II protocol to group mapping

<0x0600-0xffff>

Ether type

llc

Delete protocol-based VLAN LLC protocol to group mapping

<0x00-0xff>

DSAP value

<0x00-0xff>

SSAP value

snap

Delete protocol-based VLAN SNAP protocol to group mapping

<oui-address>

OUI address, format : 00-40-c7

<0x0000-0xffff>

Protocol ID is the Ethernet type field value for the protocol running on top of SNAP

vlan

Delete protocol-based VLAN group to VLAN mapping

<WORD>

Up to 16 characters to describe protocol-based VLAN group name

Example:

```
Switch(vcl)# delete mac-vlan 00-00-00-00-00-11
Switch(vcl)# delete protocol-vlan vlan david
```

51.2 *mac-vlan*

Using this command you can Configure MAC-based VLAN membership.

Syntax:

```
mac-vlan <mac-address> <1-4094> <port-list>
```

Parameter:**<mac-address>**

MAC address, format 0a-1b-2c-3d-4e-5f

<1-4094>

VLAN ID, possible values from 1 to 4094.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(vcl)# mac-vlan 0a-1b-2c-3d-4e-5f 4094 2
Switch(vcl)# show mac-config
MAC Address      VID    Ports
-----
0a-1b-2c-3d-4e-5f 4094   2
```

51.3 *protocol-vlan*

Using this command you can Configure protocol-based VLAN.

Syntax:

```
protocol-vlan protocol ethernet <0x0600-0xffff> <vgroup>
protocol-vlan protocol llc <0x00-0xff> <0x00-0xff> <vgroup>
protocol-vlan protocol snap <oui-address> <0x0000-0xffff>
protocol-vlan vlan <vgroup> <1-4094> <port-list>
```

Parameter:**protocol**

Protocol-based VLAN ethertype protocol to group mapping

ethernet

Protocol-based VLAN Ethernet-II protocol to group mapping

<0x0600-0xffff>

Ether type

<vgroup>

Up to 16 characters to describe protocol-based VLAN group name

llc

Protocol-based VLAN LLC protocol to group mapping

<0x00-0xff>

DSAP value

<0x00-0xff>

SSAP value

<vgroup>

Up to 16 characters to describe protocol-based VLAN group name

snap

Protocol-based VLAN SNAP protocol to group mapping

<oui-address>

OUI address, format : 00-40-c7

<0x0000-0xffff>

Protocol ID is the Ethernet type field value for the protocol running on top of SNAP

vlan

Protocol-based VLAN group to VLAN mapping

<vgroup>

Up to 16 characters to describe protocol-based VLAN group name

<1-4094>

VLAN ID, possible values from 1 to 4094.

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(vcl)# protocol-vlan protocol Ethernet 0xFFFF david
Switch(vcl)# show protocol-vlan
Protocol Type  Protocol (Value)      Group Name
-----
Ethernet      ETYPE:0xffff           david

Switch(vcl)# protocol-vlan protocol snap 00-10-cc 0xeeee kevin
Switch(vcl)# show protocol-vlan
Protocol Type  Protocol (Value)      Group Name
-----
SNAP          OUI-00:10:cc; PID:0xeeee kevin
Ethernet      ETYPE:0xffff           david

Switch(vcl)# protocol-vlan vlan jack 3000 1
Switch(vcl)# show protocol-vlan
Protocol Type  Protocol (Value)      Group Name
-----
SNAP          OUI-00:10:cc; PID:0xeeee kevin
Ethernet      ETYPE:0xffff           david

Group Name      VID  Ports
-----
jack            3000  1
```

51.4 show

Using this command you can Show VCL status command.

Syntax:

```
show mac-config
show mac-status (combined|nas|static)
show protocol-vlan
```

Parameter:**mac-config**

Shows MAC-based VLAN entry

mac-status

Shows MAC-based VLAN status

combined

Shows all the combined VCL MAC-based VLAN database

nas

Shows the VCL MAC-based VLAN configured by NAS

static

Shows the VCL MAC-based VLAN entries configured by the administrator

protocol-vlan

Shows protocol-based VLAN configuration

Example:

```
Switch(vcl)# show mac-config
MAC Address      VID    Ports
-----
00-00-00-00-00-00 3      5,6
00-00-00-00-00-11 1      1,2
00-00-00-00-00-22 2      3,4
00-00-00-00-00-33 1      2,3

Switch(vcl)# show mac-status combined
MAC Address      VID    Ports
-----
0a-1b-2c-3d-4e-5f 4094   2

Switch(vcl)# show protocol-vlan
Protocol Type    Protocol (Value)      Group Name
-----
SNAP             OUI-00:10:cc; PID:0xeeee  kevin
Ethernet         ETYPE:0xffff          david

Group Name      VID    Ports
-----
jack            3000   1
```

52 VLAN Commands

To assign a specific VLAN for management purpose. The management VLAN is used to establish an IP connection to the switch from a workstation connected to a port in the VLAN. This connection supports a VSM, SNMP, and Telnet session. By default, the active management VLAN is VLAN 1. Only one management VLAN can be active at a time.

Command	Function
delete	Delete VLAN group
egress-rule	Configure egress-rule of switch ports
forbidden	Configure forbidden VLAN group
frame-type	Configure frame type of switch ports
ingress-filtering	Configure ingress filtering of switch ports
port-type	Configure port type of switch ports
pvid	Configure port VLAN ID
show	Show VLAN information
tag-group	Configure tag-based VLAN group
tpid	Configure the TPID used for Custom S-ports. This is a global setting for all the Custom S-ports

52.1 *delete*

Using this command you can Delete VLAN group.

Syntax:

```
delete forbidden|group <1-4094>
```

Parameter:

forbidden

Delete VLAN forbidden group

group

Delete tag-based VLAN group

<1-4094>

VLAN ID, possible values from 1 to 4094.

Example:

```
Switch(vlan)# delete forbidden 1
Switch(vlan)# delete group 1
```

52.2 *egress-rule*

Using this command you can Configure egress-rule of switch ports.

Syntax:

```
egress-rule <port-list> access|hybrid|trunk
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

access

Untag all frames

hybrid

Tag all frames except VLAN ID same as PVID

trunk

Tag all frames

Example:

```
Switch(vlan)# egress-rule 1 access
Switch(vlan)# egress-rule 2 hybrid
Switch(vlan)# egress-rule 3 trunk
Switch(vlan)# show port-config
TPID for Custom S-port : 0x88a
```

Port	PVID	Frame Type	Ingress Filter	Egress Rule	Port Type
1	1	All	Disabled	Access	UnAware
2	1	All	Disabled	Hybrid	UnAware
3	1	All	Disabled	Trunk	UnAware

52.3 *forbidden*

Using this command you can Configure forbidden VLAN group.

Syntax:

```
forbidden <1-4094> <WORD> <port-list>
```

Parameter:

<1-4094>

VLAN ID, possible values from 1 to 4094.

<WORD>

Up to 33 characters describing VLAN name

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(vlan)# forbidden 1 david 2-5
Switch(vlan)# show forbidden
```

VID	VLAN Name	Ports
1	david	2-5

52.4 frame-type

Using this command you can Configure frame type of switch ports.

Syntax:

```
frame-type <port-list> all|tagged|untagged
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

all

Accept all frames

tagged

Accept tagged frames only

untagged

Accept untagged frames only

Example:

```
Switch(vlan)# frame-type 1 tagged
Switch(vlan)# frame-type 2 untagged
Switch(vlan)# show port-config
TPID for Custom S-port : 0x88a8
```

Port	PVID	Frame Type	Ingress Filter	Egress Rule	Port Type
1	1	Tagged	Disabled	Access	UnAware
2	1	Untagged	Disabled	Hybrid	UnAware
3	1	All	Disabled	Trunk	UnAware

52.5 *ingress-filtering*

Using this command you can Configure ingress filtering of switch ports.

Syntax:

```
ingress-filtering <port-list> disable|enable
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disables ingress filtering

enable

Enables ingress filtering. If ingress port is not a member of the classified VLAN of the frame, the frame is discarded

Example:

```
Switch(vlan)# ingress-filtering 1 enable
Switch(vlan)# show port-config
TPID for Custom S-port : 0x88a8
```

Port	PVID	Frame Type	Ingress Filter	Egress Rule	Port Type
1	1	Tagged	Enabled	Access	UnAware
2	1	Untagged	Disabled	Hybrid	UnAware
3	1	All	Disabled	Trunk	UnAware

52.6 *port-type*

Using this command you can Configure port type of switch ports.

Syntax:

```
port-type <port-list> c-port|s-custom-port|s-port|unaware
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

c-port

Customer port

s-custom-port

Custom Service port

s-port

Service port

unaware

VLAN unaware port

Example:

```
Switch(vlan)# port-type 2 c-port
Switch(vlan)# port-type 3 s-port
Switch(vlan)# port-type 4 s-custom-port
Switch(vlan)# show port-config
TPID for Custom S-port : 0x88a8
```

Port	PVID	Frame Type	Ingress Filter	Egress Rule	Port Type
1	1	Tagged	Enabled	Access	UnAware
2	1	Untagged	Disabled	Hybrid	C-Port
3	1	All	Disabled	Trunk	S-Port
4	1	All	Disabled	Hybrid	S-Custom-Port

52.7 pvid

Using this command you can Configure port VLAN ID.

Syntax:

```
pvid <port-list> <1-4094>
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

<1-4094>

VLAN ID, possible values from 1 to 4094.

Example:

```
Switch(vlan)# pvid 1 4000
Switch(vlan)# show port-config
TPID for Custom S-port : 0x88a8
```

Port	PVID	Frame Type	Ingress Filter	Egress Rule	Port Type
1	4000	All	Disabled	Hybrid	UnAware
2	1	All	Disabled	Hybrid	UnAware
3	1	All	Disabled	Hybrid	UnAware

52.8 *show*

Using this command you can Show VLAN information.

Syntax:

```
show forbidden|port-config  
show port-status (combined|gvrp|mstp|mvr|nas|static|voice)  
show vlan [combined|gvrp|mstp|mvr|nas|static|vcl|voice]]
```

Parameter:

forbidden

Shows VLAN forbidden group

port-config

Shows VLAN port configuration

port-status

Show VLAN port status

combined

VLAN port status for combined VLAN Users

gvrp

VLAN port status for GVRP

mstp

VLAN port status for MSTP

mvr

VLAN port status for MVR

nas

VLAN port status for NAS

static

Static VLAN port status

voice

VLAN port status for Voice VLAN

vlan

Shows VLAN group

combined

Shows all the combined VLAN database

gvrp

Shows the VLANs configured by GVRP

mstp

Shows the VLANs configured by MSTP

mvr

Shows the VLANs configured by MVR

nas

Shows the VLANs configured by NAS

static

Shows the VLAN entries configured by the administrator

vcl

Shows the VLANs configured by VCL

voice

Shows the VLANs configured by Voice VLAN

Example:

```
Switch(vlan)# show port-config
TPID for Custom S-port : 0x88a8
```

Port	PVID	Frame Type	Ingress Filter	Egress Rule	Port Type
1	1	All	Disabled	Hybrid	UnAware
2	1	All	Disabled	Hybrid	UnAware
3	1	All	Disabled	Hybrid	UnAware

```
Switch(vlan)# show port-status combined
```

Port	PVID	Frame Type	Ingress Filter	Tx Tag	UVID	Port Type	Conflict
1	1	All	Disabled	Untag This	1	UnAware	No
2	1	All	Disabled	Untag This	1	UnAware	No
3	1	All	Disabled	Untag This	1	UnAware	No

```
Switch(vlan)# show vlan combined
```

VID	VLAN Name	User	Ports
1	default	Combined	1-26

52.9 tag-group

Using this command you can Configure tag-based VLAN group.

Syntax:*tag-group* <1-4094> <name> <port-list>**Parameter:****<1-4094>**

VLAN ID, possible values from 1 to 4094.

<name>

Up to 33 characters describing VLAN name

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

Example:

```
Switch(vlan)# tag-group 3000 david 2
Switch(vlan)# show vlan
```

VID	VLAN Name	User	Ports
1	default	Static	1-26
3000	david	Static	2

52.10 *tpid*

Using this command you can Configure the TPID used for Custom S-ports. This is a global setting for all the Custom S-ports.

Syntax:

```
tpid <0x0600-0xffff>
```

Parameter:**<0x0600-0xffff>**

Configure TPID value, available value is from 0x600 to 0xffff

Example:

```
Switch(vlan)# tpid 0xffff
Switch(vlan)# show port-config
TPID for Custom S-port : 0xffff
```

Port	PVID	Frame Type	Ingress Filter	Egress Rule	Port Type
1	1	All	Disabled	Hybrid	UnAware
2	1	All	Disabled	Hybrid	UnAware
3	1	All	Disabled	Hybrid	UnAware

53 Voice VLAN Commands

Voice VLAN is VLAN configured specially for voice traffic. By adding the ports with voice devices attached to voice VLAN, we can perform QoS-related configuration for voice data, ensuring the transmission priority of voice traffic and voice quality.

The Voice VLAN feature enables voice traffic forwarding on the Voice VLAN, then the switch can classify and schedule network traffic. It is recommended that there be two VLANs on a port - one for voice, one for data. Before connecting the IP device to the switch, the IP phone should configure the voice VLAN ID correctly.

Command	Function
config	Configure Voice VLAN
delete	Delete commands
discovery	Configure Voice VLAN discovery protocol
oui	Create Voice VLAN OUI entry. Modify OUI table will restart auto detect OUI process
port-mode	Configure Voice VLAN port mode
security	Configure Voice VLAN port security mode
show	Show Voice VLAN information

53.1 *config*

Using this command you can Configure Voice VLAN.

Syntax:

```
config disable  
config enable <1-4094> <10-1000000> <0-7>
```

Parameter:

disable

Disable Voice VLAN mode operation.

enable

Enables Voice VLAN mode operation.

<1-4094>

VLAN ID, possible values from 1 to 4094..

<10-1000000>

Voice VLAN secure aging time, available value is from 10 to 1000000.

<0-7>

Voice VLAN traffic class, all traffic on the Voice VLAN will apply this class, available value is from 0(Low) to 7(High).

Example:

```
Switch(voice-vlan)# config enable 2 8888 7
Switch(voice-vlan)# show config
Voice VLAN Mode           : Enabled
Voice VLAN VLAN ID        : 2
Voice VLAN Age Time(seconds) : 8888
Voice VLAN Traffic Class   : 7

Port  Mode      Security  Discovery Protocol
-----
1     Disabled   Disabled  OUI
2     Disabled   Disabled  OUI
3     Disabled   Disabled  OUI
```

53.2 *delete*

Using this command you can Delete command.

Syntax:

```
delete oui <oui-address>
```

Parameter:**oui**

Delete Voice VLAN OUI entry. Modify OUI table will restart auto detect OUI process.

<oui-address>

OUI address, format : 0a-1b-2c.

Example:

```
Switch(voice-vlan)# delete oui 0a-1b-2c
```

53.3 *discovery*

Using this command you can Configure Voice VLAN discovery protocol.

Syntax:

```
discovery <port-list> both|lldp|oui
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

both

Both OUI and LLDP.

lldp

Detect telephony device by LLDP.

oui

Delete Voice VLAN OUI entry. Modify OUI table will restart auto detect OUI process.

Example:

```
Switch(voice-vlan)# discovery 2 both
Switch(voice-vlan)# discovery 3 lldp
Switch(voice-vlan)# show config
Voice VLAN Mode           : Enabled
Voice VLAN VLAN ID        : 2
Voice VLAN Age Time(seconds) : 8888
Voice VLAN Traffic Class   : 7

Port  Mode      Security  Discovery Protocol
----  -
1     Disabled   Disabled  OUI
2     Disabled   Disabled  Both
3     Disabled   Disabled  LLDP
```

53.4 oui

Using this command you can Create Voice VLAN OUI entry. Modify OUI table will restart auto detect OUI process.

Syntax:

```
oui <oui-address> [<description>]
```

Parameter:**<oui-address>**

OUI address, format : 0a-1b-2c.

<description>

Up to 32 characters describing OUI address.

Example:

```
Switch(voice-vlan)# oui 0a-1b-2c david
Switch(voice-vlan)# show oui
No  Telephony OUI  Description
--  -
1   00-01-E3      Siemens AG phones
2   00-03-6B      Cisco phones
3   00-0F-E2      H3C phones
4   00-60-B9      Philips and NEC AG phones
5   00-D0-1E      Pingtel phones
6   00-E0-75      Polycom phones
7   00-E0-BB      3Com phones
8   0A-1B-2C      david
```

53.5 *port-mode*

Using this command you can Configure Voice VLAN port mode.

Syntax:

```
port-mode <port-list> auto|disable|force
```

Parameter:

<port-list>

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

auto

Enables auto detect mode. It detects whether there is VoIP phone attached on the specific port and configure the Voice VLAN members automatically.

disable

Disjoin from Voice VLAN

force

Forced join to Voice VLAN

Example:

```
Switch(voice-vlan)# port-mode 1 auto
Switch(voice-vlan)# port-mode 2 force
Switch(voice-vlan)# show config
Voice VLAN Mode           : Enabled
Voice VLAN VLAN ID        : 2
Voice VLAN Age Time(seconds) : 8888
Voice VLAN Traffic Class   : 7

Port  Mode      Security  Discovery Protocol
----  -
1     Auto       Disabled OUI
2     Forced     Disabled OUI
3     Disabled   Disabled OUI
```

53.6 *security*

Using this command you can Configure Voice VLAN port security mode.

Syntax:

```
security <port-list> disable|enable
```

Parameter:**<port-list>**

Port list, possible values depending on distinct hardware model. Individual ports are separated by comma, port ranges are joined by hyphen (1,3-5).

disable

Disjoin from Voice VLAN

enable

Enables Voice VLAN security mode. When the function is enabled, all non-telephone MAC address in Voice VLAN will be blocked 10 seconds

Example:

```
Switch(voice-vlan)# security 1 enable
Switch(voice-vlan)# show config
Voice VLAN Mode           : Enabled
Voice VLAN VLAN ID        : 2
Voice VLAN Age Time(seconds) : 8888
Voice VLAN Traffic Class   : 7

Port  Mode      Security  Discovery Protocol
----  -
1     Disabled   Enabled   OUI
2     Disabled   Disabled  OUI
3     Disabled   Disabled  OUI
```

53.7 show

Using this command you can Show Voice VLAN information.

Syntax:

```
show config|oui
```

Parameter:**config**

Shows Voice VLAN configuration.

oui

Shows OUI address

Example:

```
Switch(voice-vlan)# show config
Voice VLAN Mode           : Disabled
Voice VLAN VLAN ID        : 1000
Voice VLAN Age Time(seconds) : 86400
Voice VLAN Traffic Class   : 7

Port  Mode      Security  Discovery Protocol
----  -
1     Disabled   Disabled  OUI
```

```
2      Disabled Disabled OUI
3      Disabled Disabled OUI

Switch(voice-vlan)# show oui
No  Telephony OUI  Description
--  -
1   00-01-E3      Siemens AG phones
2   00-03-6B      Cisco phones
3   00-0F-E2      H3C phones
4   00-60-B9      Philips and NEC AG phones
5   00-D0-1E      Pingtel phones
6   00-E0-75      Polycom phones
7   00-E0-BB      3Com phones
```

54 Global Commands

The Global commands is probably the most commonly used in the CLI console. It is used for global configuration at any level of command.

Command	Function
auto-logout	Configure time of inactivity before automatic logout
exit	Exit from current mode
help	Show available commands
history	Show a list of previously run commands
logout	Disconnect
quit	Disconnect
restore	Restore running configuration
save	Save running configuration

54.1 *auto-logout*

Using this command you can Configure time of inactivity before automatic logout.

Syntax:

```
auto-logout <10-3600>
```

Parameter:

<10-3600>

Time in seconds of inactivity before automatic logout.

Example:

```
Switch# auto-logout 3600
```

54.2 *exit*

Using this command you can Exit from current mode.

Syntax:

```
exit
```

Example:

```
Switch(aaa)# exit
Switch#
```

54.3 *help*

This command displays the CLI help.

Syntax:

```
help
?
```

Parameter:**help**

Displays command help (available commands and description).

?

Displays parameter help (available commands and description).

Example:

```
Switch# help

Commands available:
  aaa             Authentication, Authorization, Accounting
  access          Access management
  account         User account management
  ...             ...
  vlan            VLAN configuration
  voice-vlan      Voice VLAN configuration

Global commands:
  auto-logout     Configure time of inactivity before automatic logout

  exit            Exit from current mode
  help            Show available commands
  history         Show a list of previously run commands
  logout          Disconnect
  restore         Restore running configuration
  save            Save running configuration

Switch# auto-logout ?
  <10-3600>       Time in seconds of inactivity before automatic
logout
```

54.4 *history*

Using this command you can Show a list of previously run commands.

Syntax:*history***Example:**

```
Switch# history
Command history:
 0. help
 1. history
 2. 0
 3. history
 4. 3
 5. history
```

54.5 *logout*

Using this command you can Disconnect.

Syntax:*logout***Example:**

```
Switch# logout
Username:
```

54.6 *restore*

Using this command you can Restore running configuration.

Syntax:

```
restore default [keep-ip]
restore user
```

Parameter:**default**

Restore configuration as factory default.

user

Restore configuration as user configuration.

keep-ip

Restore configuration as factory default unless ip address.

Example:

```
Switch# restore default keep-ip  
Switch# restore user
```

54.7 *save*

Using this command you can Save running configuration.

Syntax:

```
save start|user
```

Parameter:**start**

Save running configuration as start configuration.

user

Save running configuration as user configuration.

Example:

```
Switch# restore default keep-ip  
Switch# restore user
```